A lowest-order Stabilization-free Virtual Element Method for 3D problems

Computer Methods in Applied Mechanics and Engineering 462 (2026) 119322

Contents lists available at ScienceDirect

Comput. Methods Appl. Mech. Engrg.

journal homepage: www.elsevier.com/locate/cma

A lowest-order Stabilization-free Virtual Element Method for 3D problems

Stefano Berrone, Andrea Borio, Francesca Marcon *, Fabio Vicini

Politecnico di Torino, Department of Mathematical Sciences "Giuseppe Luigi Lagrange", Corso Duca degli Abruzzi 24, Torino, 10129, Italy
INdAM research group GNCS member, Italy

ARTICLE INFO

MSC:
65N12
65N30
74G15
74S99

Keywords:
Virtual element methods
Stabilization-free
3D problems
Elasticity

ABSTRACT

This work aims at presenting a novel Stabilization-free Virtual Element method for 3D problems. In particular, we focus on the discretization of the linear elastic equation, but the new projection operator introduced can be applied to derive a self-stabilized formulation also for general scalar elliptic equations. The method is introduced in its lowest order formulation and for the analysis we consider the class of polyhedra with triangular faces, typically called deltahedra. We provide a sufficient condition on the polynomial projection space that implies the well-posedness. Several numerical tests assess the robustness of the method and confirm the theoretical convergence rates. Furthermore, we test the proposed method on a nonlinear elasticity problem, to show the robustness of the proposed self-stabilized formulation for solving nonlinear problems.

1. Introduction

In recent years, numerical methods for the approximation of partial differential equations on polygonal and polyhedral meshes have attracted considerable attention within the scientific community, thanks to their great flexibility in handling highly complex geometries. Among the various approaches proposed in this framework, the Virtual Element Method (VEM) represents one of the most widely used families of polytopal discretization techniques. Since its introduction in the seminal work [1], VEMs have been extensively applied to a wide range of engineering applications. The key feature of virtual elements lies in the use of implicitly defined basis functions, which are not explicitly computed, together with a suitable set of degrees of freedom from which polynomial projections can be constructed. In classical VEM formulations, these projections are involved in the definition of the discrete bilinear forms, requiring the addition of an arbitrary non-polynomial stabilizing term, to ensure the coercivity of these forms. The stabilization term requires to be tuned according to the problem under consideration, which constitutes a limitation, especially for problems characterized by strong anisotropies and nonlinearities.

To overcome this drawback, novel VEM formulations that do not require a stabilization term, referred to as Stabilization-Free VEM (SFVEM), have been recently introduced [2–6] and have attracted considerable attention within the scientific community. These novel formulations allow for the definition of self-stabilized bilinear forms, relying only on consistent and stable polynomial projections, preserving the structure of the underlying operator. These methods have been applied in various contexts, including problems characterized by anisotropy and convection-dominated regimes [7,8], problems of interest in computational mechanics [9–11], as well as the development of a posteriori error estimators [12]. However, only a few contributions address three-dimensional

* Corresponding author at: Politecnico di Torino, Department of Mathematical Sciences "Giuseppe Luigi Lagrange", Corso Duca degli Abruzzi 24, Torino, 10129, Italy.

E-mail addresses: stefano.berrone@polito.it (S. Berrone), andrea.borio@polito.it (A. Borio), francesca.marcon@polito.it (F. Marcon), fabio.vicini@polito.it (F. Vicini).

https://doi.org/10.1016/j.cma.2026.119322
Received 22 May 2026; Received in revised form 22 July 2026; Accepted 15 August 2026

Available online 28 August 2026

applications [13–15], and the theoretical development of three-dimensional formulations is still in its early stages. Meanwhile, we also mention other approaches that aim to overcome some issues related to the stabilization of polygonal methods, such as [16], where a reduced-basis method is proposed to design the stabilization term, [17–19], where bounds for the stability terms used in the a posteriori error analysis are presented, [20], where neural networks are integrated to approximate the basis functions, and [21,22], where other stabilization-free approaches are proposed in the context of Hybrid High-Order methods.

In this work, we propose a new framework for the development of SFVEM on three-dimensional meshes, extending the approach based on divergence-free projections developed for the 2D case in [23,24]. We consider the linear elasticity equation in primal form and investigate the stability of a new projection operator, that applies from the space of gradients of 3D VEM functions to the space of divergence-free polynomials. As a preliminary step for the analysis, we consider the class of polyhedra with triangular faces, typically called deltahedra, and a low order formulation. We identify a sufficient condition on the local polynomial projection space that ensures the coercivity of the new gradient projection. Then, we show that applying the proposed polynomial projection to the discretization of the linear elasticity equation yields a well-posed formulation. We remark that the proposed approach can be straightforwardly applied to general second order elliptic equations. Since the polynomial degree of the proposed projection depends on the geometry of the polyhedron, we propose an algorithm based on an optimized QR decomposition, as already done for the 2D case in [23,24]. Then, we devise a wide campaign of tests on general deltahedra to numerically verify that the proposed scheme is stable and robust, together with standard convergence tests on different meshes, which confirm the expected rates of convergence. Furthermore, we apply the proposed method for solving nonlinear elasticity problems, with a comparison with the standard VEM discretization. The objective is to show that the proposed SFVEM method exhibits a stable convergence of the Newton method thanks to the definition of a self-stabilized and operator-preserving discretization. In particular, it displays the expected quadratic convergence, allowing the residual threshold to be reached in fewer iterations. Consequently, the proposed method results to be more robust for the solution of nonlinear problems, as it does not require the tuning of any non-physical parameter, and significantly reduces the computational cost and the overall efficiency.

This work is organized as follows. In Section 2 we state our model problem and in Section 3 we present the new projection operator and the discrete scheme. The well-posedness of the discrete formulation is discussed in Section 4 and Section 5 shows the computation of the new projector. Finally, Section 6 contains the numerical results discussed above.

2. Model problem

Let $\Omega \subset \mathbb{R}^3$ be a bounded open set. We are interested in solving the following problem:

$$\begin{cases} -\nabla \cdot \boldsymbol{\sigma}(\boldsymbol{u}) = \boldsymbol{f} & \text{in } \Omega, \\ \boldsymbol{u} = 0 & \text{on } \partial\Omega. \\ \boldsymbol{\sigma}(\boldsymbol{u}) = D\boldsymbol{\varepsilon}(\boldsymbol{u}) = 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) + \lambda \operatorname{tr}(\boldsymbol{\varepsilon}(\boldsymbol{u}))\boldsymbol{I}\,, & \\ \boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2}\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^{\intercal}\right)\,, & \end{cases} \tag{1}$$

where μ and λ are the Lamé coefficients, $\boldsymbol{I}$ represents the second order identity tensor, and D is the fourth order elasticity tensor. For the sake of simplicity, we restrict our analysis to homogeneous Dirichlet boundary conditions. Extending the results to more general boundary conditions involves additional standard technicalities. We consider the following standard regularity assumptions for μ and λ (see for instance [25]): there exists $\mu_{\min}, \kappa_{\min} > 0$ such that

$$\mu, \lambda \in \mathrm{L}^{\infty}(\Omega)\,, \quad \mu(\boldsymbol{x}) \geq \mu_{\min}\,, \quad \lambda(\boldsymbol{x}) + \frac{2}{3}\mu(\boldsymbol{x}) \geq \kappa_{\min}\,, \quad \text{a.e. } \boldsymbol{x} \in \Omega. \tag{2}$$

To obtain the variational formulation of problem (1), let $\mathbf{V} := [\mathrm{H}^1_0(\Omega)]^3$. Moreover, let $a : \mathbf{V} \times \boldsymbol{V} \to \mathbf{R}$ such that,

$$a\left(\boldsymbol{u}, \boldsymbol{v}\right) = \int_{\Omega} \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) = \int_{\Omega} D\boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) \quad \forall \boldsymbol{u}, \boldsymbol{v} \in \mathbf{V}\,. \tag{3}$$

Notice that, under assumption (2), there exists $\eta_{\min} > 0$ such that for any suitable $\boldsymbol{u}$

$$a\left(\boldsymbol{u}, \boldsymbol{u}\right) \geq \eta_{\min} \int_{\Omega} \boldsymbol{\varepsilon}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{u}) = \eta_{\min} \|\boldsymbol{\varepsilon}(\boldsymbol{u})\|^2_{\mathrm{L}^2(\Omega)} \tag{4}$$

Then, given $\boldsymbol{f} \in [\mathrm{L}^2(\Omega)]^3$, the variational formulation of (1) is: find $\boldsymbol{u} \in \mathbf{V}$ such that,

$$a\left(\boldsymbol{u}, \boldsymbol{v}\right) = (\boldsymbol{f}, \boldsymbol{v})_{\Omega} \quad \forall \boldsymbol{v} \in \mathbf{V}\,, \tag{5}$$

where $(\cdot,\cdot)_{\Omega}$ denotes the standard L^2 scalar product.

Let $\mathbf{V}$ be endowed with the norm

$$\|\boldsymbol{v}\|^2_{\mathbf{V}} = \int_{\Omega} \nabla \boldsymbol{v} : \nabla \boldsymbol{v}.$$

It can be proved by suitable Korn's inequalities that there exist two constants $M, \alpha > 0$, depending on Ω, μ, and λ, such that

$$\alpha \|\boldsymbol{v}\|^2_{\mathbf{V}} \leq a\left(\boldsymbol{v}, \boldsymbol{v}\right) \leq M \|\boldsymbol{v}\|^2_{\mathbf{V}}.$$

The above inequalities guarantee the well-posedness of problem (5).

3. Discrete formulation

In this section, we propose a stabilization-free virtual element discretization of (5) defined on polyhedral meshes with triangular faces. First, we introduce the scalar virtual space and the new projection operator, and then we embed it in a vector-valued framework. Let $\mathcal{M}_h$ denote a conforming polyhedral decomposition of the domain Ω, and $E \in \mathcal{M}_h$ denotes a generic polyhedra. We assume that $\mathcal{M}_h$ satisfies the following assumptions: there exists a positive constant ρ such that

A.1 for all $E \in \mathcal{M}_h$, E is star-shaped with respect to a sphere of radius $\rho \geq \rho h_E$, where h_E is the diameter of E;
A.2 each face f of E is star-shaped with respect to a ball of radius $\eta \geq \rho h_f$, where h_f is the diameter of f;
A.3 for each face f of E and for each edge e of f, $|e| \geq \rho h_f \geq \rho^2 h_E$.

For any given $\omega \subset \Omega$, let $\mathbb{P}_k(\omega)$ denote the space of polynomials of degree up to k defined on ω. Hence, let $\Pi_1^{\nabla,\omega} : \mathrm{H}^1(\omega) \to \mathbb{P}_1(\omega)$ be the $\mathrm{H}^1(\omega)$-orthogonal projection defined up to a constant by the orthogonality condition:

$$\forall u \in \mathrm{H}^1(\omega), \quad \left(\nabla\left(\Pi_1^{\nabla,\omega} u - u\right), \nabla p\right)_\omega = 0 \quad \forall p \in \mathbb{P}_1(\omega). \tag{6}$$

In the following, for any $E \in \mathcal{M}_h$ and for any face $f \subset \partial E$, we use $\Pi_1^{\nabla,f}$ and $\Pi_1^{\nabla,E}$, respectively. In order to define these projectors uniquely, we set

$$\int_{\partial f} \gamma^{\partial f}\left(\Pi_1^{\nabla,f} u\right) = \int_{\partial f} \gamma^{\partial f}(u) \ \forall f \subset \partial E \quad \text{and} \quad \int_{\partial E} \gamma^{\partial E}\left(\Pi_1^{\nabla,E} u\right) = \sum_{f \subset \partial E} \int_f \Pi_1^{\nabla,f} u. \tag{7}$$

In the following, for the sake of simplicity we will omit the trace operator γ when it is clear from the integration domain. For any given $E \in \mathcal{M}_h$, let $\mathcal{V}_{h,1}^E$ be the local scalar Virtual Element Space of order 1:

$$\begin{aligned}\mathcal{V}_{h,1}^E &:= \{v_h \in \mathrm{H}^1(E) : \ \Delta v_h \in \mathbb{P}_1(E), \gamma^f\left(v_h\right) \in \mathcal{V}_{h,1}^f \ \forall f \subset \partial E, \gamma^{\partial E}\left(v_h\right) \in C^0(\partial E), \\ &\left(v_h - \Pi_1^{\nabla,E} v_h, p\right)_E = 0 \ \forall p \in \mathbb{P}_1(E)\}.\end{aligned} \tag{8}$$

where, for any face f

$$\begin{aligned}\mathcal{V}_{h,1}^f &:= \{v_h \in \mathrm{H}^1(f) : \ \Delta v_h \in \mathbb{P}_1(f), \ \gamma^e\left(v_h\right) \in \mathbb{P}_1(e) \ \forall e \subset \partial f, \ \gamma^{\partial f}\left(v_h\right) \in C^0(\partial f), \\ &\left(v_h - \Pi_1^{\nabla,f} v_h, p\right)_f = 0 \ \forall p \in \mathbb{P}_1(f)\}.\end{aligned}$$

We recall that the degrees of freedom of $\mathcal{V}_{h,1}^E$ are the values of functions at the vertices of E (see [26,27]). Now, let us prove a property, derived from [28], that will be useful in the following.

Lemma 1. *There exists a positive constant C, independent of h_E, such that for any $v_h \in \mathcal{V}_{h,1}^E$*

$$|v_h|_{\mathrm{H}^1(E)}^2 \leq C h_E \left\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\right\|_{\mathrm{L}^2(\partial E)}^2, \tag{9}$$

where $\wedge$ denotes the cross product and $\boldsymbol{n}^{\partial E}$, with a slight abuse of notation, denotes the outward-pointing unit normal vector to the boundary of E, i.e. it collects the different unit normal vectors $\boldsymbol{n}^f$ associated with each face.

Proof. Let us recall a known result for virtual functions [28]: $\forall v_h \in \mathcal{V}_{h,1}^E$ we have

$$|v_h|_{\mathrm{H}^1(E)}^2 \lesssim h_E^{-1} \sum_{f \subset \partial E} \left\|\Pi_0^{0,f} v_h\right\|_{\mathrm{L}^2(f)}^2 + h_E \left\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\right\|_{\mathrm{L}^2(\partial E)} \tag{10}$$

where $\lesssim$ denotes that there exists a constant independent of h_E and $\Pi_0^{0,f}$ denotes the L^2-orthogonal projector on $\mathbb{P}_0(f)$. Let $\mathrm{P}_0(v_h) := \frac{1}{|\partial E|} \int_{\partial E} v_h$, applying (10), the continuity of the L^2-orthogonal projector and a Poincaré inequality, we get

$$\begin{aligned}|v_h|_{\mathrm{H}^1(E)}^2 &= |v_h - \mathrm{P}_0(v_h)|_{\mathrm{H}^1(E)}^2 \lesssim h_E^{-1} \sum_{f \subset \partial E} \left\|\Pi_0^{0,f}(v_h - \mathrm{P}_0(v_h))\right\|_{\mathrm{L}^2(f)}^2 + h_E \left\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\right\|_{\mathrm{L}^2(\partial E)}^2 \\ &\lesssim h_E^{-1} \|v_h - \mathrm{P}_0(v_h)\|_{\mathrm{L}^2(\partial E)}^2 + h_E \left\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\right\|_{\mathrm{L}^2(\partial E)}^2 \lesssim h_E \left\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\right\|_{\mathrm{L}^2(\partial E)}^2. \quad \square\end{aligned} \tag{11}$$

Let us propose a new projection operator for the gradient of a virtual function. To achieve this, for any given $E \in \mathcal{M}_h$, let $\ell_E \in \mathbb{N}$ be given. We will discuss in the following how to choose ℓ_E. Now, we consider the following decomposition of $\left[\mathbb{P}_{\ell_E+1}(E)\right]^3$ (see [29])

$$\left[\mathbb{P}_{\ell_E+1}(E)\right]^3 = \nabla \mathbb{P}_{\ell_E+2}(E) \oplus \mathcal{G}_{\ell_E+1}^{\perp}(E), \tag{12}$$

where

$$\mathcal{G}_{\ell_E+1}^{\perp}(E) = \boldsymbol{x} \wedge \left[\mathbb{P}_{\ell_E}(E)\right]^3. \tag{13}$$

Notice that $\nabla\mathbb{P}_{\ell_E+2}(E)$ is the kernel of the **curl** operator in $\left[\mathbb{P}_{\ell_E+1}(E)\right]^3$, hence

$$\mathbf{curl}\left[\mathbb{P}_{\ell_E+1}(E)\right]^3 = \mathbf{curl}\,\mathcal{G}^{\perp}_{\ell_E+1}(E)\,, \tag{14}$$

where for any $\boldsymbol{p} \in \left[\mathbb{P}_{\ell_E+1}(E)\right]^3$, i.e. $\boldsymbol{p} = (p_x, p_y, p_z)$,

$$\mathbf{curl}\,\boldsymbol{p} = \left(\frac{\partial p_z}{\partial y} - \frac{\partial p_y}{\partial z}\,,\, \frac{\partial p_x}{\partial z} - \frac{\partial p_z}{\partial x}\,,\, \frac{\partial p_y}{\partial x} - \frac{\partial p_x}{\partial y}\right)^{\intercal}. \tag{15}$$

Let $\hat{\Pi}^{0,E}_{\ell_E}\nabla : \mathrm{H}^1(E) \to \mathbf{curl}\,\mathcal{G}^{\perp}_{\ell_E+1}(E)$ be the L^2-projector of the gradient of functions in $\mathrm{H}^1(E)$ defined by the following orthogonality condition: $\forall u \in \mathrm{H}^1(E)$

$$\left(\hat{\Pi}^{0,E}_{\ell_E}\nabla u, \mathbf{curl}\,\boldsymbol{p}\right)_E = (\nabla u, \mathbf{curl}\,\boldsymbol{p})_E \quad \forall \boldsymbol{p} \in \mathcal{G}^{\perp}_{\ell_E+1}(E)\,. \tag{16}$$

Remark 1. We remark that

$$\mathbf{curl}\,\mathcal{G}^{\perp}_{\ell_E+1}(E) = \{\boldsymbol{p} \in \left[\mathbb{P}_{\ell_E}(E)\right]^3 : \nabla\cdot\boldsymbol{p} = 0\}\,, \tag{17}$$

holds true, based on the exactness – i.e. *the image of every operator coincides with the kernel of the following one* (see [30])– of the following sequence in three dimensions

$$\mathbb{P}_{\ell_E+2}(E) \xrightarrow{\nabla} \left[\mathbb{P}_{\ell_E+1}(E)\right]^3 \xrightarrow{\mathbf{curl}} \left[\mathbb{P}_{\ell_E}(E)\right]^3 \xrightarrow{\nabla\cdot} \mathbb{P}_{\ell_E-1}(E)\,, \tag{18}$$

together with (14).

Remark 2. Considering the previous relation, we can see that if $u \in \mathbb{P}_1(E)$, then $\hat{\Pi}^{0,E}_{\ell_E}\nabla u = \nabla u$.

Moreover, applying the product rule for the divergence of the cross product of two vector fields in three dimensions, known results about De Rham diagrams in Sobolev spaces and the Green-Gauss formula, we get $\forall u \in \mathrm{H}^1(E)$ and $\forall \boldsymbol{p} \in \left[\mathbb{P}_{\ell_E+1}(E)\right]^3$

$$(\nabla u, \mathbf{curl}\,\boldsymbol{p})_E = \int_E \nabla\cdot(\boldsymbol{p}\wedge\nabla u) = \int_{\partial E}(\boldsymbol{p}\wedge\nabla u)\cdot\boldsymbol{n}^{\partial E} = \int_{\partial E}(\nabla u\wedge\boldsymbol{n}^{\partial E})\cdot\boldsymbol{p}\,. \tag{19}$$

Proposition 2. *Let E be such that all the faces $f \subset \partial E$ are triangles. Then, $\hat{\Pi}^{0,E}_{\ell_E}\nabla v$ is computable for any $v \in \mathcal{V}^E_{h,1}$.*

Proof. For any $f \subset \partial E$, since f is a triangle, then $\mathcal{V}^f_{h,1} \equiv \mathbb{P}_1(f)$ and $\nabla u \wedge \boldsymbol{n}^f \in \left[\mathbb{P}_0(f)\right]^3$. Hence, considering (19) it is easy to see that the projector $\hat{\Pi}^{0,E}_{\ell_E}\nabla$ is computable. □

From this point onward, we restrict our analysis to the case in which $\mathcal{M}_h$ consists only of polyhedra with triangular faces. Then, for any function $\boldsymbol{u} = (u_x, u_y, u_z) \in [\mathrm{H}^1(E)]^3$, let $\hat{\Pi}^{0,E}_{\ell_E}\nabla\boldsymbol{u}$ be defined such that

$$\hat{\Pi}^{0,E}_{\ell_E}\nabla\boldsymbol{u} = \left(\hat{\Pi}^{0,E}_{\ell_E}\nabla u_x\,,\quad \hat{\Pi}^{0,E}_{\ell_E}\nabla u_y\,,\quad \hat{\Pi}^{0,E}_{\ell_E}\nabla u_z\right)^{\intercal}, \tag{20}$$

and the local discrete counterpart of $\boldsymbol{\varepsilon}(\boldsymbol{u})$, denoted by $\boldsymbol{\varepsilon}^E_h(\boldsymbol{u})$ is defined as

$$\boldsymbol{\varepsilon}^E_h(\boldsymbol{u}) = \frac{1}{2}\left(\hat{\Pi}^{0,E}_{\ell_E}\nabla\boldsymbol{u} + \left(\hat{\Pi}^{0,E}_{\ell_E}\nabla\boldsymbol{u}\right)^{\intercal}\right). \tag{21}$$

Let the local bilinear form $a^E_h : [\mathrm{H}^1(E)]^3 \times [\mathrm{H}^1(E)]^3 \to \mathbb{R}$ be defined as

$$a^E_h(\boldsymbol{u},\boldsymbol{v}) := \int_E D\boldsymbol{\varepsilon}^E_h(\boldsymbol{u}) : \boldsymbol{\varepsilon}^E_h(\boldsymbol{v}) \quad \forall \boldsymbol{u},\boldsymbol{v} \in [\mathrm{H}^1(E)]^3\,, \tag{22}$$

and the global bilinear form $a_h : \mathbf{V}\times\mathbf{V} \to \mathbb{R}$ as

$$a_h(\boldsymbol{u},\boldsymbol{v}) := \sum_{E\in\mathcal{M}_h} a^E_h(\boldsymbol{u},\boldsymbol{v}) \quad \forall \boldsymbol{u},\boldsymbol{v} \in \mathbf{V}\,. \tag{23}$$

Let $\boldsymbol{\mathcal{V}}^E_{h,1}$ be the vector-valued virtual element space, i.e. $\boldsymbol{\mathcal{V}}^E_{h,1} = [\mathcal{V}^E_{h,1}]^3$, then we can state the discrete problem as: find $\boldsymbol{u}_h \in \boldsymbol{\mathcal{V}}_{h,1}$, where

$$\boldsymbol{\mathcal{V}}_{h,1} := \{\boldsymbol{v} \in \mathbf{V} : \boldsymbol{v}_{|E} \in \boldsymbol{\mathcal{V}}^E_{h,1}\}\,, \tag{24}$$

such that

$$a_h(\boldsymbol{u}_h,\boldsymbol{v}_h) = \sum_{E\in\mathcal{M}_h}\left(\boldsymbol{f}, \Pi^{0,E}_1\boldsymbol{v}_h\right)_E \quad \forall \boldsymbol{v}_h \in \boldsymbol{\mathcal{V}}_{h,1}\,, \tag{25}$$

where, given the scalar L^2-orthogonal projection operator on linear polynomials $\Pi_1^{0,E} : \mathrm{L}^2(E) \to \mathbb{P}_1(E)$, i.e. for each $v \in \mathrm{L}^2(E)$

$$\left(\Pi_1^{0,E} v - v, p\right)_E = 0 \ \forall\, p \in \mathbb{P}_1(E) \,, \tag{26}$$

computable by standard VEM techniques [31], the vector-valued counterpart is given by

$$\Pi_1^{0,E} \boldsymbol{v}_h = \left(\Pi_1^{0,E}(v_h)_x \,, \quad \Pi_1^{0,E}(v_h)_y \,, \quad \Pi_1^{0,E}(v_h)_z\right)^{\intercal} . \tag{27}$$

4. Well-posedness

In this section, we want to prove the well-posedness of the discrete formulation stated in (25), defined on a general polyhedral mesh with triangular faces. To this end, we show some preliminary results on the scalar virtual space and the projection $\hat{\Pi}_{\ell_E}^{0,E}\nabla$ defined in (16). In particular, in Theorem 3 we show that the projector $\hat{\Pi}_{\ell_E}^{0,E}\nabla$ induces a seminorm on the local virtual space, equivalent with the local H^1-seminorm. To prove it, the following assumption on ℓ_E is required.

H.1 ℓ_E is such that there exists a set of degrees of freedom for $\boldsymbol{p} \in \mathcal{G}_{\ell_E+1}^{\perp}(E)$ which contains the integral mean $\frac{1}{|f|}\int_f \boldsymbol{p}$ on each face $f \subset \partial E$

Theorem 3. *Under the assumption A.1, A.2, A.3 and H.1, if E is such that all the faces $f \subset \partial E$ are triangles, then there exists a positive constant β_*^E, independent of h_E, such that*

$$\left\|\hat{\Pi}_{\ell_E}^{0,E}\nabla v_h\right\|_{\mathrm{L}^2(E)} \geq \beta_*^E \left|v_h\right|_{\mathrm{H}^1(E)} \quad \forall v_h \in \mathcal{V}_{h,1}^E \,. \tag{28}$$

Proof. Using the definition of the operator $\hat{\Pi}_{\ell_E}^{0,E}\nabla v$, (19) and that for any face $f \subset \partial E$ $\mathcal{V}_{h,1}^f = \mathbb{P}_1(f)$, we get $\forall v_h \in \mathcal{V}_{h,1}^E$

$$\begin{aligned}\left\|\hat{\Pi}_{\ell_E}^{0,E}\nabla v_h\right\|_{\mathrm{L}^2(E)} &= \sup_{\boldsymbol{p}\in\mathcal{G}_{\ell_E+1}^{\perp}(E)} \frac{\left(\nabla v_h, \mathbf{curl}\,\boldsymbol{p}\right)_E}{\|\mathbf{curl}\,\boldsymbol{p}\|_{\mathrm{L}^2(E)}} = \sup_{\boldsymbol{p}\in\mathcal{G}_{\ell_E+1}^{\perp}(E)} \frac{\int_{\partial E}(\nabla v_h \wedge \boldsymbol{n}^{\partial E})\cdot \boldsymbol{p}}{\|\mathbf{curl}\,\boldsymbol{p}\|_{\mathrm{L}^2(E)}} \\ &= \sup_{\boldsymbol{p}\in\mathcal{G}_{\ell_E+1}^{\perp}(E)} \frac{\sum_{f\subset\partial E}(\nabla v_h \wedge \boldsymbol{n}^f)\cdot\int_f \boldsymbol{p}}{\|\mathbf{curl}\,\boldsymbol{p}\|_{\mathrm{L}^2(E)}}\end{aligned} \tag{29}$$

Under assumption H.1, we define $\tilde{\boldsymbol{p}} \in \mathcal{G}_{\ell_E+1}^{\perp}(E)$ such that its degrees of freedom are

$$\frac{1}{|f|}\int_f \tilde{\boldsymbol{p}} = h_E^2(\nabla v_h \wedge \boldsymbol{n}^f) \quad \forall f \subset \partial E \tag{30}$$

and all other degrees of freedom of $\tilde{\boldsymbol{p}}$ are zero. Since $\mathcal{G}_{\ell_E+1}^{\perp}(E) \cap \ker(\mathbf{curl}\,(\cdot)) = \{\boldsymbol{0}\}$ (see (14)), $\|\mathbf{curl}\,\boldsymbol{p}\|_{\mathrm{L}^2(E)}$ is a norm of $\mathcal{G}_{\ell_E+1}^{\perp}(E)$. Hence, applying the equivalence of norms between $\|\mathbf{curl}\,\boldsymbol{p}\|_{\mathrm{L}^2(E)}$ and the l^2-norm of the vector of degrees of freedom of $\tilde{\boldsymbol{p}}$, the definition of $\tilde{\boldsymbol{p}}$ and assumption A.3, we get the estimate

$$\begin{aligned}\|\mathbf{curl}\,\tilde{\boldsymbol{p}}\|_{\mathrm{L}^2(E)}^2 \leq C_p^2 h_E \sum_{f\subset\partial E}\left(\frac{1}{|f|}\int_f \tilde{\boldsymbol{p}}\right)^2 &= C_p^2 h_E \sum_{f\subset\partial E}\frac{h_E^4}{|f|}\int_f (\nabla v_h \wedge \boldsymbol{n}^f)^2 \\ &\leq \frac{C_p^2}{\kappa^2} h_E^3 \left\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\right\|_{\mathrm{L}^2(\partial E)}^2 .\end{aligned} \tag{31}$$

With the above definition of $\tilde{\boldsymbol{p}}$, starting from (29) and applying (31), we obtain

$$\begin{aligned}\left\|\hat{\Pi}_{\ell_E}^{0,E}\nabla v_h\right\|_{\mathrm{L}^2(E)} = \sup_{\boldsymbol{p}\in\mathcal{G}_{\ell_E+1}^{\perp}(E)} \frac{\sum_{f\subset\partial E}(\nabla v_h \wedge \boldsymbol{n}^f)\cdot\int_f \boldsymbol{p}}{\|\mathbf{curl}\,\boldsymbol{p}\|_{\mathrm{L}^2(E)}} &\geq \frac{\kappa}{C_p}\frac{h_E^2\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\|_{\mathrm{L}^2(\partial E)}^2}{h_E^{\frac{3}{2}}\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\|_{\mathrm{L}^2(\partial E)}} \\ &= \frac{\kappa}{C_p} h_E^{\frac{1}{2}}\left\|\nabla v_h \wedge \boldsymbol{n}^{\partial E}\right\|_{\mathrm{L}^2(\partial E)} .\end{aligned} \tag{32}$$

Finally, applying Lemma 1 we get the thesis. ☐

Remark 3 (*Scalar Elliptic Problems*)**.** The result stated in Theorem 3 implies that the stabilization-free VEM discretization of the 3D Poisson problem: given $f \in \mathrm{L}^2(\Omega)$, find $u_h \in \mathcal{V}_{h,1}^E$

$$\sum_{E\in\mathcal{M}_h}\left(\hat{\Pi}_{\ell_E}^{0,E}\nabla u_h, \hat{\Pi}_{\ell_E}^{0,E}\nabla v_h\right)_E = \sum_{E\in\mathcal{M}_h}\left(f, \Pi_1^{0,E} v_h\right)_E \ \forall\, v_h \in \mathcal{V}_{h,1}^E \tag{33}$$

is well-posed, applying the same techniques as in [23]. The proposed method can also be straightforwardly extended to the 3D diffusion–advection–reaction problem, following the approach described in [24,27].

In order to prove the continuity and the coercivity of the bilinear form a_h, we need to prove a discrete *curl*-type Korn's inequality, stated in Proposition 5, for the space $[\mathcal{G}_s^{\perp}(E)]^{3r}$, that denotes the space of tensors obtained patching row-vectors of $\mathcal{G}_s^{\perp}(E)$, for any given degree $s \geq 0$. Hence, for any $\pi \in [\mathcal{G}_s^{\perp}(E)]^{3r}$ we apply row-wise the **curl** operator and we define the symmetric **curl** operator as

$$\mathbf{curl}^{\,S}\pi := \frac{1}{2}\left(\mathbf{curl}\,\pi + (\mathbf{curl}\,\pi)^{\intercal}\right) . \tag{34}$$

To prove Proposition 5, we need the following preliminary result.

Lemma 4. *The quantity $\|\mathbf{curl}^{\,S}\pi\|_{\mathrm{L}^2(E)}$ is a norm for any $\pi \in [\mathcal{G}_s^{\perp}(E)]^{3r}$.*

Proof. This proof is based on the argument presented in [32]. Assume that $\mathbf{curl}^{\,S}\pi = 0$. Since $\pi \in [\mathcal{G}_s^{\perp}(E)]^{3r}$, by (14) $\mathbf{curl}\,\pi = 0$ if and only if $\pi = 0$. Hence, $\mathbf{curl}^{\,S}\pi = 0$ if and only if $\mathbf{curl}\,\pi$ is an anti-symmetric matrix. Assume that

$$\mathbf{curl}\,\pi = A \tag{35}$$

where

$$A = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \tag{36}$$

for some $\boldsymbol{a} = (a_x, a_y, a_z) \in \left[\mathbb{P}_{s-1}(E)\right]^3$. Since $\nabla \cdot \mathbf{curl}\,\pi = 0$, then

$$\nabla \cdot A = -\mathbf{curl}\,\boldsymbol{a} = 0 . \tag{37}$$

The previous relation, applying (14), is equivalent to $\boldsymbol{a} = -\nabla q$ for some polynomial $q \in \mathbb{P}_s(E)$. A simple computation yields $\mathbf{curl}\,(q\boldsymbol{I}) = A = \mathbf{curl}\,\pi$, thus

$$\pi = q\boldsymbol{I} . \tag{38}$$

Since $\pi \in [\mathcal{G}_s^{\perp}(E)]^{3r}$, by (13) there exists $\boldsymbol{p}_1, \boldsymbol{p}_2, \boldsymbol{p}_3 \in \mathbb{P}_{s-1}(E)$ such that

$$\pi = \begin{bmatrix} \boldsymbol{x} \wedge \boldsymbol{p}_1 \\ \boldsymbol{x} \wedge \boldsymbol{p}_2 \\ \boldsymbol{x} \wedge \boldsymbol{p}_3 \end{bmatrix} . \tag{39}$$

Then, by (38),

$$\begin{cases} \boldsymbol{x} \wedge \boldsymbol{p}_1 = (q, 0, 0) \\ \boldsymbol{x} \wedge \boldsymbol{p}_2 = (0, q, 0) \\ \boldsymbol{x} \wedge \boldsymbol{p}_3 = (0, 0, q) \end{cases} , \tag{40}$$

Hence, $\boldsymbol{x}$ has to be orthogonal to three linearly independent vectors. This is true if and only if $q = 0$, and thus $\pi = 0$. □

Proposition 5. *For any $\pi \in [\mathcal{G}_s^{\perp}(E)]^{3r}$ there exists $C_*^E > 0$, independent of h_E, such that*

$$C_*^E \|\mathbf{curl}\,\pi\|_{\mathrm{L}^2(E)} \leq \left\|\mathbf{curl}^{\,S}\pi\right\|_{\mathrm{L}^2(E)} . \tag{41}$$

Proof. Let the space $[\mathcal{G}_s^{\perp}(E)]^{3r}$ be endowed with the following two norms: for any $\pi \in [\mathcal{G}_s^{\perp}(E)]^{3r}$

$$\|\pi\|_1 := h_E^{-1}\|\pi\|_{\mathrm{L}^2(E)} + \|\mathbf{curl}\,\pi\|_{\mathrm{L}^2(E)} . \tag{42}$$

and $\|\pi\|_2 := \left\|\mathbf{curl}^{\,S}\pi\right\|_{\mathrm{L}^2(E)}$, as proved in Lemma 4. Then, it follows by equivalence of norms on finite dimensional spaces and a suitable scaling argument that there exists $C > 0$ independent of h_E such that for any $\pi \in [\mathcal{G}_s^{\perp}(E)]^{3r}$

$$h_E^{-1}\|\pi\|_{\mathrm{L}^2(E)} + \|\mathbf{curl}\,\pi\|_{\mathrm{L}^2(E)} \leq C\left\|\mathbf{curl}^{\,S}\pi\right\|_{\mathrm{L}^2(E)} . \tag{43}$$

The thesis (41) follows immediately. □

Theorem 6. *The discrete bilinear form a_h defined by (23) satisfies that there $\exists \alpha^*$ independent of h such that*

$$a_h\left(\boldsymbol{u}_h, \boldsymbol{v}_h\right) \leq \alpha^* \|\boldsymbol{u}_h\|_{\mathbf{V}} \|\boldsymbol{v}_h\|_{\mathbf{V}} \quad \forall \boldsymbol{u}_h, \boldsymbol{v}_h \in \mathcal{V}_{h,1} . \tag{44}$$

Then, assuming that **H.1** *is satisfied for each $E \in \mathcal{M}_h$, there $\exists \alpha_*$ independent of h such that*

$$a_h\left(\boldsymbol{v}_h, \boldsymbol{v}_h\right) \geq \alpha_* \|\boldsymbol{v}_h\|_{\mathbf{V}}^2 \quad \forall \boldsymbol{v}_h \in \mathcal{V}_{h,1} . \tag{45}$$

Proof. The proof of (44) can be derived immediately applying the definition of a_h (23), the regularity assumption $\mu, \lambda \in \mathrm{L}^\infty(\Omega)$ (2), the Cauchy–Schwarz inequality and the continuity of the operator $\hat{\Pi}^{0,E}_{\ell_E}$, that can be derived directly from its definition applying the Cauchy–Schwarz inequality, i.e.

$$
\begin{aligned}
a_h\left(\boldsymbol{u}_h, \boldsymbol{v}_h\right) &= \sum_{E \in \mathcal{M}_h} \int_E D \boldsymbol{\varepsilon}_h^E(\boldsymbol{u}_h) : \boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h) \leq \alpha^* \sum_{E \in \mathcal{M}_h} \left\| \boldsymbol{\varepsilon}_h^E(\boldsymbol{u}_h) \right\|_{\mathrm{L}^2(E)} \left\| \boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h) \right\|_{\mathrm{L}^2(E)} \\
&\leq \alpha^* \sum_{E \in \mathcal{M}_h} \left\| \hat{\Pi}^{0,E}_{\ell_E} \nabla \boldsymbol{u}_h \right\|_{\mathrm{L}^2(E)} \left\| \hat{\Pi}^{0,E}_{\ell_E} \nabla \boldsymbol{v}_h \right\|_{\mathrm{L}^2(E)} \leq \alpha^* \|\boldsymbol{u}_h\|_{\mathbf{V}} \|\boldsymbol{v}_h\|_{\mathbf{V}} .
\end{aligned} \tag{46}
$$

where $\alpha^* = 2\|\mu\|_\infty + \|\lambda\|_\infty$.

Applying the definition of a_h (23) and noticing that (4) holds true for a_h, we get

$$
a_h\left(\boldsymbol{v}_h, \boldsymbol{v}_h\right) = \sum_{E \in \mathcal{M}_h} a_h^E\left(\boldsymbol{v}_h, \boldsymbol{v}_h\right) \geq \eta_{\min} \sum_{E \in \mathcal{M}_h} \left\| \boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h) \right\|^2_{\mathrm{L}^2(E)} \geq \eta_{\min} C_* \sum_{E \in \mathcal{M}_h} \left\| \hat{\Pi}^{0,E}_{\ell_E} \nabla \boldsymbol{v}_h \right\|^2_{\mathrm{L}^2(E)} , \tag{47}
$$

where in the last step we apply Proposition 5 with $C_* = \min_{E \in \mathcal{M}_h} C_*^E$, noticing that for any $\boldsymbol{v}_h$, by definition of $\hat{\Pi}^{0,E}_{\ell_E} \nabla \boldsymbol{v}_h$, there exists $\pi_{\boldsymbol{v}_h} \in [\mathcal{G}^\perp_{\ell_E+1}(E)]^{3r}$ such that $\hat{\Pi}^{0,E}_{\ell_E} \nabla \boldsymbol{v}_h = \mathbf{curl}\, \pi_{\boldsymbol{v}_h}$. Hence, $\boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h) = \mathbf{curl}^S(\pi_{\boldsymbol{v}_h})$. Finally, applying Theorem 3, we get the thesis (45) with $\alpha_* := \eta_{\min} C_* \min_{E \in \mathcal{M}_h} \beta_*^E$. □

Lastly, we remark that Theorem 6 implies that a_h defined in (23) satisfies the hypothesis of the Lax–Milgram theorem, hence the discrete problem (25) admits a unique solution. Then, we observe that a_h satisfies the assumptions of the abstract convergence theorem stated in [33]. Hence, the classical a priori error estimates hold true, as reminded in the following theorem.

Theorem 7. *Let $\boldsymbol{u} \in [\mathrm{H}^2(\Omega) \cap \mathrm{H}^1_0(\Omega)]^3$ be the exact solution of (5). Then $\exists C > 0$ independent of h such that the unique solution $\boldsymbol{u}_h \in \mathcal{V}_{h,1}$ of (25) satisfies the following estimates:*

$$
\|\boldsymbol{u} - \boldsymbol{u}_h\|_{\mathbf{V}} \leq C h \left(|\boldsymbol{u}|_2 + \|\boldsymbol{f}\|_{\mathrm{L}^2(\Omega)} \right) , \tag{48}
$$

$$
\|\boldsymbol{u} - \boldsymbol{u}_h\|_{\mathrm{L}^2(\Omega)} \leq C h^2 \left(|\boldsymbol{u}|_2 + \|\boldsymbol{f}\|_{\mathrm{L}^2(\Omega)} \right) . \tag{49}
$$

5. Computation of the projection operator $\hat{\Pi}^{0,E}_{\ell_E} \nabla$

Algorithm 1 Algorithm for the computation of ℓ_E and of the corresponding right-hand side B on a given polyhedron.

Input: A polyhedron $E \in \mathcal{M}_h$, a chosen threshold *tol*
- Let ℓ_E be the smallest number satisfying (53).
- Compute the matrix B corresponding to ℓ_E defined by (50).
- Compute the factor R of a QR decomposition of B.
- $N \leftarrow$ number of diagonal elements of R whose absolute value is $\geq tol$.
- **while** $N < N_E - 1$ **do**
    - $\ell_E \leftarrow \ell_E + 1$.
    - Compute $\tilde{B}$ such that $\tilde{B}_{ij} = \sum_{f \subset \partial E} \left(\nabla \varphi_j \wedge \boldsymbol{n}^f, \boldsymbol{m}_i \right)_f$ $\forall \boldsymbol{m}_i$ basis of $\mathcal{G}^\perp_{\ell_E+1}(E) / \mathcal{G}^\perp_{\ell_E}(E)$
    - Using Givens rotations, update R adding the rows of $\tilde{B}$.
    - $B \leftarrow \begin{bmatrix} B & \tilde{B} \end{bmatrix}^\intercal$.
    - $N \leftarrow$ number of diagonal elements of R whose absolute value is $\geq tol$.
- **end while**
- **return** ℓ_E, B.

In this section, we analyze the computation of the scalar projector $\hat{\Pi}^{0,E}_{\ell_E} \nabla$ defined by (16) and, following [23,24], we propose an algorithm that provides the polynomial projection degree ensuring stability. The algorithm is based on an incremental QR decomposition of the right-hand side of the linear system that needs to be solved to compute the matrix representing the projection operator. In particular, let $E \in \mathcal{M}_h$ be given, let N_E be the number of vertices of E and let $\{\varphi_j\}_{j=1}^{N_E}$ be the Lagrange basis of $\mathcal{V}^E_{h,1}$ with respect to the degrees of freedom. Furthermore, given any value $\ell_E \in \mathbb{N}$, let $\{\boldsymbol{m}_i\}_{i=1}^{\dim \mathcal{G}^\perp_{\ell_E+1}(E)}$ be a basis of the polynomial space $\mathcal{G}^\perp_{\ell_E+1}(E)$. See for instance [34] for a the construction of a particular basis. Let B be the matrix given by

$$
B_{ij} = \sum_{f \subset \partial E} \left(\nabla \varphi_j \wedge \boldsymbol{n}^f, \boldsymbol{m}_i \right)_f \quad \forall i = 1, \dots, \dim \mathcal{G}^\perp_{\ell_E+1}(E), \; j = 1, \dots, N_E . \tag{50}
$$

Hence, the matrix $\hat{\Pi}$ representing the projection defined in (16) is the solution of the matrix equation

$$
G \hat{\Pi} = B , \tag{51}
$$

where G is the mass matrix of the set $\{\mathbf{curl}\, \boldsymbol{m}_i\}_{i=1}^{\dim \mathcal{G}^\perp_{\ell_E+1}(E)}$. In order to get a stable scheme, we need to ensure that the rank of $\hat{\Pi}$ is at least $N_E - 1$, and since G is symmetric and positive definite, this is guaranteed if the rank of B is $N_E - 1$. Following [23,24],

Table 1
Regular Polyhedra. Values of ℓ_E^0 and ℓ_E given by the necessary condition (53) and by Algorithm 1, respectively, and the corresponding scaled singular values.

E	N_E	ℓ_E^0	ℓ_E	h_E	$\sigma_{\max}/\sqrt{h_E}$	$\sigma_{\min}/\sqrt{h_E}$
Tet	4	0	0	$1.41 \cdot 10^0$	$5.77 \cdot 10^{-1}$	$2.89 \cdot 10^{-1}$
Hexa	8	1	2	$1.73 \cdot 10^0$	$5.52 \cdot 10^{-1}$	$3.15 \cdot 10^{-1}$
Octahedron	6	1	1	$2.00 \cdot 10^0$	$4.08 \cdot 10^{-1}$	$3.23 \cdot 10^{-1}$
Icosahedron	12	1	2	$2.00 \cdot 10^0$	$3.98 \cdot 10^{-1}$	$3.79 \cdot 10^{-1}$
Dodecahedron	20	2	3	$2.00 \cdot 10^0$	$4.40 \cdot 10^{-1}$	$1.20 \cdot 10^{-1}$
UV sphere	30	3	3	$2.00 \cdot 10^0$	$4.22 \cdot 10^{-1}$	$1.04 \cdot 10^{-3}$

we employ an incremental Q-less QR decomposition of B, with increasing polynomial degrees, which is described in Algorithm 1. First, we recall that

$$\dim \left[\mathbb{P}_{\ell_E+1}(E)\right]^3 = \frac{(\ell_E+2)(\ell_E+3)(\ell_E+4)}{2}\,, \text{ and } \dim \nabla\mathbb{P}_{\ell_E+2}(E) = \frac{(\ell_E+4)(\ell_E+5)(\ell_E+6)}{6} - 1\,. \tag{52}$$

Then, we set ℓ_E to be the smallest integer satisfying

$$\begin{aligned}\dim \mathcal{G}^{\perp}_{\ell_E+1}(E) \geq \dim \nabla\mathcal{V}^E_{h,1} \iff& \dim\left[\mathbb{P}_{\ell_E+1}(E)\right]^3 - \dim\nabla\mathbb{P}_{\ell_E+2}(E) \geq \dim\nabla\mathcal{V}^E_{h,1}\\ \iff& (\ell_E+1)(\ell_E+2)(2\ell_E+9) \geq 6(N_E-1)\,,\end{aligned} \tag{53}$$

which is a necessary condition for the injectivity of $\hat{\Pi}^{0,E}_{\ell_E}\nabla$. We then compute the B matrix corresponding to ℓ_E and the factor R of an Householder QR-decomposition of B. The rank of B is then computed as the number of diagonal elements of R that are greater than a chosen threshold *tol* (that we chose equal to 10^{-10}). If the rank is not large enough, we compute additional rows of the right-hand side corresponding to the basis of $\mathcal{G}^{\perp}_{\ell_E+1}(E)/\mathcal{G}^{\perp}_{\ell_E}(E)$ and we update the R matrix by using Givens rotations on the additional rows, until we reach the desired rank. Notice that the computational cost of the algorithm is equivalent to the one of computing the R matrix of a QR decomposition of the final right-hand side. Applying Algorithm 1, we identify the minimum ℓ_E providing numerically the coercivity and the corresponding right-hand side B. Then, we compute the matrix G and solve (51).

Remark 4. We emphasize that the value of ℓ_E depends only on the geometry of the considered polyhedron. Consequently, it can computed only once for each element $E \in \mathcal{M}_h$. This feature becomes particularly advantageous when iterative procedures are adopted, since the associated computational cost is incurred only during the first iteration, hence significantly mitigating its impact on the overall computational effort (see, for example, the nonlinear framework discussed in Section 6.4). Furthermore, in the next section, we show through several polyhedral test cases that the numerical sufficient value required to guarantee local stability is consistently very close to the necessary value given by (53). In most cases, the two values coincide or the sufficient value exceeds the necessary one by only a single unit.

6. Numerical results

In this section, we investigate the performance of the proposed SFVEM method. We first carry out a numerical investigation of the local stability of the proposed method, applying Algorithm 1 on different set of polyhedra with triangular faces to discretize the local contribution of the scalar 3D Poisson problem, detailed in Remark 3. Then, we present a convergence test on a linear elastic problem to assess the robustness of the proposed method and confirm the expected theoretical convergence rates. Finally, we apply the proposed method to solve a nonlinear problem, comparing its behavior with a standard VEM discretization, proposed for 2D problems in [35]. All numerical results are obtained using the open-source C++ library PolyDiM, see [36].

6.1. Coercivity tests

We consider different polyhedra and we apply Algorithm 1 to compute the value ℓ_E and get a stable local discrete bilinear form. For each polyhedron, we build the local stiffness matrix A^E related to the local discrete bilinear form of the scalar 3D Poisson problem, detailed in Remark 3, i.e.

$$A^E_{j,k} = \left(\hat{\Pi}^{0,E}_{\ell_E}\nabla\varphi_j, \hat{\Pi}^{0,E}_{\ell_E}\nabla\varphi_k\right)_E, \quad \forall j,k = 1,\dots,N_E,$$

where $\{\varphi_j\}_{j=1}^{N_E}$ is the Lagrange basis of $\mathcal{V}^E_{h,1}$ with respect to the degrees of freedom. In all the cases, we compute the square root of the largest and the second smallest singular value, denoted respectively by $\sigma_{\max}$ and $\sigma_{\min}$. Since the matrix A^E in 3D scales as h_E, we check if $\sigma_{\min}/h_E$ is well detached from zero, ensuring that the rank of the matrix is $N_E - 1$.

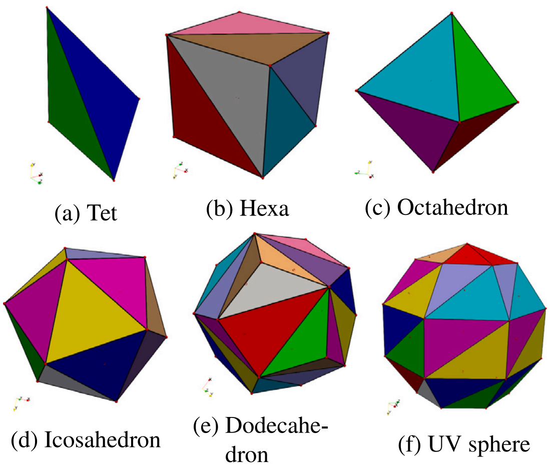

(a) Tet (b) Hexa (c) Octahedron
(d) Icosahedron (e) Dodecahedron (f) UV sphere

Fig. 1. Regular polyhedra used for tests.

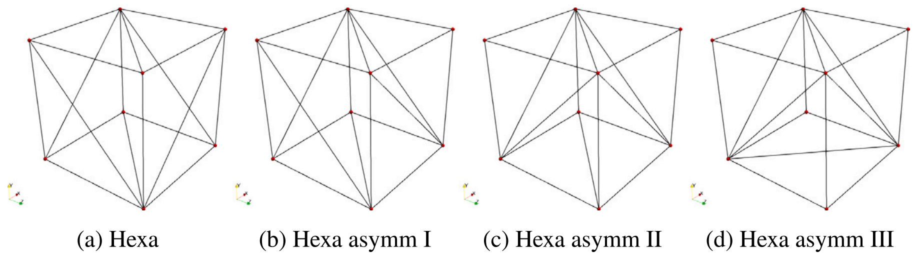

(a) Hexa (b) Hexa asymm I (c) Hexa asymm II (d) Hexa asymm III

Fig. 2. Hexahedron case - different symmetries of faces triangulation.

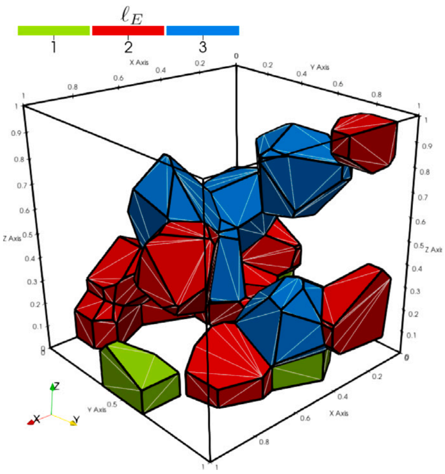
Table 2
Hexahedron. Values of ℓ_E^0 and ℓ_E given by the necessary condition (53) and by Algorithm 1, respectively, and the corresponding scaled singular values.

E	ℓ_E^0	ℓ_E	$\sigma_{\max}/\sqrt{h_E}$	$\sigma_{\min}/\sqrt{h_E}$
Hexa	1	2	$5.52 \cdot 10^{-1}$	$3.15 \cdot 10^{-1}$
Hexa asymm I	1	1	$4.62 \cdot 10^{-1}$	$1.56 \cdot 10^{-1}$
Hexa asymm II	1	1	$4.55 \cdot 10^{-1}$	$1.52 \cdot 10^{-1}$
Hexa asymm III	1	2	$4.70 \cdot 10^{-1}$	$2.83 \cdot 10^{-1}$

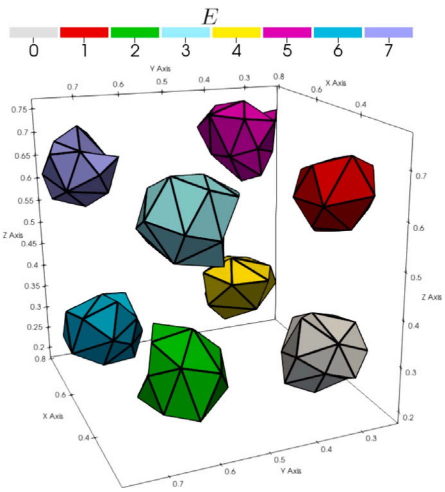
6.1.1. Regular polyhedra

We first analyze the six polyhedra shown in Fig. 1, obtained from regular polyhedra by sub-triangulating each face without altering the total number of vertices. In Table 1 we report the computed singular values, together with ℓ_E^0 and ℓ_E that are, respectively, the smallest value of ℓ_E satisfying the necessary condition (53) and the value of ℓ_E obtained applying Algorithm 1, that ensures the local stability of the corresponding A^E. We can see that $\sigma_{\min}/h_E$ is significantly larger than machine precision for all the polyhedra considered.

Furthermore, we perform a second test on a regular hexahedron. Since the definition of the projector $\hat{\Pi}_\ell^{0,E}$ is based on triangular faces, all the faces of the hexahedron are subdivided into triangles, taking care not to introduce new vertices. The choice of sub-triangulation is not unique, in Fig. 2 we depict four different configurations. The configuration in Fig. 2(a) is the symmetric case, in which the same diagonal direction is chosen on opposite faces. In contrast, Figs. 2(b)–2(d) illustrate the asymmetric cases. Table 2 reports the computed singular values together with ℓ_E^0 and ℓ_E, as in the previous test. We observe that, although all four polyhedra share the same vertices and the same number of faces, the resulting values of ℓ_E that ensure the stability of the proposed method and the resulting singular values differ depending on the case. We remark that all the singular values reported are still significantly distant from machine precision. These results suggest that the presence of symmetries may influence the value of ℓ_E that ensure the stability. This possible dependence of the sufficient value of ℓ_E on the geometry and the symmetries is a behavior already observed in the two-dimensional case, see [5].



(a) Convex polyhedra randomly selected from $\mathcal{D}_{\text{voro-poisson}}$ second mesh of the dataset in Sorgente et al. (2022), adding the sub-triangulation of faces. Each polyhedron is colored depending of the value of ℓ_E given by Algorithm 1.



(b) Concave polyhedra obtained by random agglomeration of tetrahedra.

Fig. 3. Polyhedra of general shape.

Table 3
Convex polyhedra. Values of ℓ_E^0 and ℓ_E given by the necessary condition (53) and by Algorithm 1, respectively, and the corresponding scaled singular values. The elements are ordered in descending order with respect to $\sigma_{\min}/\sqrt{h_E}$.

E	N_E	ℓ_E^0	ℓ_E	h_E	$\sigma_{\max}/\sqrt{h_E}$	$\sigma_{\min}/\sqrt{h_E}$
14	12	1	1	$3.37 \cdot 10^{-1}$	$1.26 \cdot 10^{0}$	$3.42 \cdot 10^{-3}$
37	26	2	2	$3.78 \cdot 10^{-1}$	$1.35 \cdot 10^{0}$	$8.40 \cdot 10^{-3}$
83	37	3	3	$4.56 \cdot 10^{-1}$	$1.23 \cdot 10^{0}$	$9.96 \cdot 10^{-3}$
47	24	2	2	$4.25 \cdot 10^{-1}$	$1.25 \cdot 10^{0}$	$1.02 \cdot 10^{-2}$
26	22	2	2	$4.14 \cdot 10^{-1}$	$1.15 \cdot 10^{0}$	$1.27 \cdot 10^{-2}$
29	34	3	3	$4.45 \cdot 10^{-1}$	$1.20 \cdot 10^{0}$	$3.27 \cdot 10^{-2}$
60	30	3	3	$4.58 \cdot 10^{-1}$	$1.23 \cdot 10^{0}$	$3.47 \cdot 10^{-2}$
28	21	2	2	$3.79 \cdot 10^{-1}$	$1.37 \cdot 10^{0}$	$3.53 \cdot 10^{-2}$
9	18	2	2	$3.65 \cdot 10^{-1}$	$1.23 \cdot 10^{0}$	$3.69 \cdot 10^{-2}$
13	20	2	2	$4.08 \cdot 10^{-1}$	$1.39 \cdot 10^{0}$	$3.73 \cdot 10^{-2}$
3	16	2	2	$3.06 \cdot 10^{-1}$	$1.40 \cdot 10^{0}$	$5.56 \cdot 10^{-2}$
6	19	2	2	$3.97 \cdot 10^{-1}$	$1.23 \cdot 10^{0}$	$5.71 \cdot 10^{-2}$
34	32	3	3	$4.37 \cdot 10^{-1}$	$1.27 \cdot 10^{0}$	$6.09 \cdot 10^{-2}$
44	28	3	3	$4.36 \cdot 10^{-1}$	$1.07 \cdot 10^{0}$	$6.38 \cdot 10^{-2}$
2	14	2	2	$3.22 \cdot 10^{-1}$	$1.36 \cdot 10^{0}$	$8.31 \cdot 10^{-2}$
110	17	2	2	$3.11 \cdot 10^{-1}$	$1.42 \cdot 10^{0}$	$8.81 \cdot 10^{-2}$
5	11	1	1	$3.18 \cdot 10^{-1}$	$1.39 \cdot 10^{0}$	$1.38 \cdot 10^{-1}$
17	10	1	1	$3.54 \cdot 10^{-1}$	$1.55 \cdot 10^{0}$	$1.97 \cdot 10^{-1}$
1	13	2	2	$3.22 \cdot 10^{-1}$	$1.57 \cdot 10^{0}$	$2.92 \cdot 10^{-1}$
0	8	1	2	$2.17 \cdot 10^{-1}$	$1.33 \cdot 10^{0}$	$8.00 \cdot 10^{-1}$

6.2. Polyhedra of general shape

In this subsection we analyze polyhedra of more general shape. We consider two different cases, depicted in Fig. 3. First, we focus on generic convex polyhedra, selected from the second mesh of the random Voronoi dataset $\mathcal{D}_{\text{voro-Poisson}}$ introduced in [37], with an increasing number of vertices. Fig. 3(a) shows the selected polyhedra, highlighting the aligned triangular faces that we add without introducing new vertices. Each element is colored according to the final value of ℓ_E obtained. The index of each selected mesh element is reported in column E of Table 3, together with the computed singular values and the values of ℓ_E^0 and ℓ_E. The elements are ordered in descending order with respect to the computed value $\sigma_{\min}/h_E$. We observe that, despite the geometric complexity of the polyhedra and the high presence of aligned faces, the computed value of ℓ_E coincides with the minimum value ℓ_E^0 required to satisfy the necessary condition for all elements except the last one, corresponding to $E = 0$.

Table 4
Concave polyhedra. Values of ℓ_E^0 and ℓ_E given by the necessary condition (53) and by Algorithm 1, respectively, and the corresponding scaled singular values. The elements are ordered in descending order with respect to $\sigma_{\min}/\sqrt{h_E}$.

E	N_E	ℓ_E^0	ℓ_E	h_E	$\sigma_{\max}/\sqrt{h_E}$	$\sigma_{\min}/\sqrt{h_E}$
3	17	2	2	$2.12 \cdot 10^{-1}$	$1.32 \cdot 10^{0}$	$1.03 \cdot 10^{-1}$
5	17	2	2	$2.02 \cdot 10^{-1}$	$1.35 \cdot 10^{0}$	$1.05 \cdot 10^{-1}$
0	17	2	2	$2.08 \cdot 10^{-1}$	$1.39 \cdot 10^{0}$	$1.09 \cdot 10^{-1}$
6	16	2	2	$1.97 \cdot 10^{-1}$	$1.44 \cdot 10^{0}$	$1.33 \cdot 10^{-1}$
4	16	2	2	$2.15 \cdot 10^{-1}$	$1.38 \cdot 10^{0}$	$2.27 \cdot 10^{-1}$
1	15	2	2	$2.00 \cdot 10^{-1}$	$1.39 \cdot 10^{0}$	$3.12 \cdot 10^{-1}$
2	16	2	2	$2.06 \cdot 10^{-1}$	$1.44 \cdot 10^{0}$	$3.57 \cdot 10^{-1}$
7	15	2	2	$1.80 \cdot 10^{-1}$	$1.56 \cdot 10^{0}$	$6.41 \cdot 10^{-1}$

The second analysis focuses on generic concave polyhedra, shown in Fig. 3(b). We construct eight different concave polyhedra starting from a regular tetrahedral mesh by agglomerating all the tetrahedra sharing a randomly selected vertex. Table 4 reports the computed singular values and the values of ℓ_E^0 and ℓ_E, again ordered in descending order with respect to $\sigma_{\min}/h_E$. Once more, despite the concavity and the variability in the number of faces and sizes of the polyhedra, all computed values of ℓ coincide with the minimum value ℓ_0 required to satisfy the necessary condition (53). Finally, also in these tests we observe that $\sigma_{\min}/h_E$ is well detached from zero for all the considered elements.

6.3. Convergence test for the linear elastic case

We analyze the linear elasticity problem introduced in Eq. (1) on the unit cubic domain $\Omega = [0,1]^3$, considering a material with constant coefficients $\lambda = 3.0$ and $\mu = 1.5$. We impose non-homogeneous Dirichlet boundary conditions on $\partial\Omega$ and choose the body force $\boldsymbol{f}$ such that the exact solution is given by

$$\boldsymbol{u} = [u_x, u_y, u_z] = [u,\ 1.1u,\ 1.5u]^T, \quad \text{where } u(x,y,z) := 64xyz(1-x)(1-y)(1-z) + 1.1.$$

In the following, we show the convergence curves of the L^2 and H^1 errors that we measure as

$$\begin{aligned} err_{\mathrm{L}^2} &:= \frac{1}{\|\boldsymbol{u}\|_{\mathrm{L}^2(\Omega)}} \sqrt{\sum_{E\in\mathcal{M}_h} \left\| \boldsymbol{u} - \Pi_1^{0,E}\boldsymbol{u}_h \right\|_{\mathrm{L}^2(E)}^2}, \\ err_{\mathrm{H}^1} &:= \frac{1}{\|\nabla\boldsymbol{u}\|_{\mathrm{L}^2(\Omega)}} \sqrt{\sum_{E\in\mathcal{M}_h} \left\| \nabla\boldsymbol{u} - \Pi_0^{0,E}\nabla\boldsymbol{u}_h \right\|_{\mathrm{L}^2(E)}^2} \end{aligned} \tag{54}$$

where $\boldsymbol{u}_h$ is the discrete solution to (25), obtained selecting locally ℓ_E as described previously.

Moreover, we solve the problem with the standard VEM method [35,38], where for any $\boldsymbol{u}_h, \boldsymbol{v}_h \in \mathcal{V}_{h,1}^E$ and given

$$\boldsymbol{\varepsilon}_h^E(\boldsymbol{u}_h) = \frac{1}{2}\left(\Pi_0^{0,E}\nabla\boldsymbol{u}_h + \left(\Pi_0^{0,E}\nabla\boldsymbol{u}_h\right)^{\intercal} \right), \tag{55}$$

we have

$$a_h^E\left(\boldsymbol{u}_h, \boldsymbol{v}_h\right) := \int_E D\boldsymbol{\varepsilon}_h^E(\boldsymbol{u}_h) : \boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h) + S^E\left((\mathrm{I} - \Pi_1^{\nabla,E})\boldsymbol{u}_h, (\mathrm{I} - \Pi_1^{\nabla,E})\boldsymbol{v}_h \right)$$

and $S^E : \mathcal{V}_{h,1}^E \times \mathcal{V}_{h,1}^E \to \mathbb{R}$ denotes the local *dofi-dofi* stabilizing bilinear form

$$S^E\left(\boldsymbol{u}_h, \boldsymbol{v}_h\right) = \sum_{i=\{x,y,z\}} \varsigma^E\, h_E\, \mathrm{dof}_{\mathcal{V}_{h,1}^E}\left(\boldsymbol{u}_h\right)_i \cdot \mathrm{dof}_{\mathcal{V}_{h,1}^E}\left(\boldsymbol{v}_h\right)_i \tag{56}$$

where ς^E is a *problem-scaling* constant, typically chosen for linear elastic problems as 2μ for every E, and $\mathrm{dof}_{\mathcal{V}_{h,1}^E}\left(\boldsymbol{v}_h\right)_i$ denotes the vector of degrees of freedom of the ith component of $\boldsymbol{v}_h$.

We also apply the standard first-order Finite Element Method (FEM) for solving the considered problem on tetrahedral and hexahedral meshes. We remark that the measure of the L^2 and H^1 errors defined above in (54) simplify to the standard L^2 and H^1 error for finite element functions.

We consider three sequences of meshes for the convergence test. The first sequence is made by tetrahedral meshes generated using TetGen [39], the second sequence is a tessellation made by hexahedra built using an in-house developed code, and the third sequence is a Voronoi polyhedral mesh, named $\mathcal{D}_{\text{voro-Poisson}}$, taken from the open-source dataset in [37]. For each mesh type, we consider four different and comparable cell sizes. Moreover, we remark that the faces of the polyhedra in both the hexahedral and the polyhedral meshes are sub-triangulated, as detailed in the previous section. In particular, for the hexahedral mesh we adopt the first face splitting configuration discussed in Section 6.1.1. For these meshes, we report below the results obtained using the proposed SFVEM, the standard VEM, and, where applicable, the standard FEM.

In particular, we report in Fig. 4 (solid lines) the computed L^2 and H^1 errors using (54). Clearly, the three methods behave equivalently on the family of tetrahedral meshes, confirming the validity of the in-house developed code. For all datasets, we observe

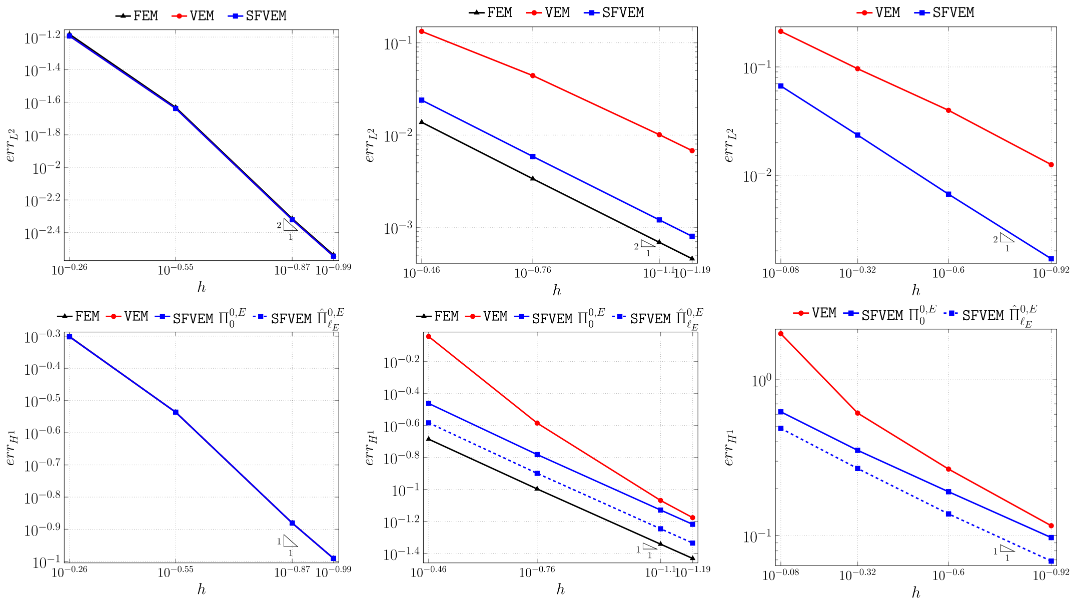


Fig. 4. Convergence plots for the linear elastic case. Results on tetrahedral meshes (left), hexahedral meshes (center) and Voronoi meshes $\mathcal{D}_{\text{voro-Poisson}}$ (right). In the H^1-error plots, the solid line refers to the related measure in (54), the dashed line to the measure (57).

that the results confirm the expected convergence rates, stated in Theorem 7. We observe that the proposed SFVEM method appears to provide a better approximation of the exact solution with respect to the standard VEM, particularly in the L^2 norm, approaching the accuracy of FEM in the hexahedral case. Furthermore, we report in Fig. 4 (dashed lines) the computed H^1 error using the following measure

$$err_{\mathrm{H}^1} := \frac{1}{\|\nabla \boldsymbol{u}\|_{\mathrm{L}^2(\Omega)}} \sqrt{\sum_{E\in\mathcal{M}_h} \left\|\nabla \boldsymbol{u} - \hat{\Pi}^{0,E}_{\ell_E} \nabla \boldsymbol{u}_h\right\|^2_{\mathrm{L}^2(E)}}, \tag{57}$$

that we propose as an alternative measure for the H^1 error made by the proposed method, that do not require additional computational cost w.r.t the one in (54). In the legend of Fig. 4, SFVEM $\Pi_0^{0,E}$ denotes the H^1-error computed according to (54), whereas SFVEM $\Pi_{\ell_E}^{0,E}$ denotes the corresponding error computed using (57). As above, we remark that the error measure (57) simplifies to the standard H^1 error for finite element functions. From Fig. 4 (dashed lines), we observe that this error measure gives an expected improved approximation of the exact solution gradients, comparable to that of FEM in the hexahedral case, while preserving an overall equivalence in behavior among all the considered methods.

Fig. 5 shows the distribution of the nodal error between the exact solution $\boldsymbol{u}$ and the discrete solution $\boldsymbol{u}_h$ on the cross-section of the third hexahedral mesh defined by the plane passing through the point $[0.55, 0.5, 0.5]$ with normal vector $[1, 0, 0]$. We observe that the FEM and SFVEM errors are of similar orders of magnitude, whereas the stabilized VEM errors are approximately one order of magnitude larger. While the error distributions do not match exactly across the considered methods, as expected, they exhibit similar overall trends.

Finally, we assess the impact of running Algorithm 1 on the overall assembly process for the finest hexahedral and polyhedral meshes. In our implementation, considering the hexahedral mesh, which consists of 19683 three-dimensional cells, the computation of ℓ_E accounts for approximately 18% of the total assembly time. On the other hand, considering the Voronoi mesh, which consists of 4941 cells, this percentage decreases to approximately 7%. We note that this cost can be significantly reduced for structured meshes such as the hexahedral mesh we use, where the projection matrix needs to be computed only once and can then be reused for all cells.

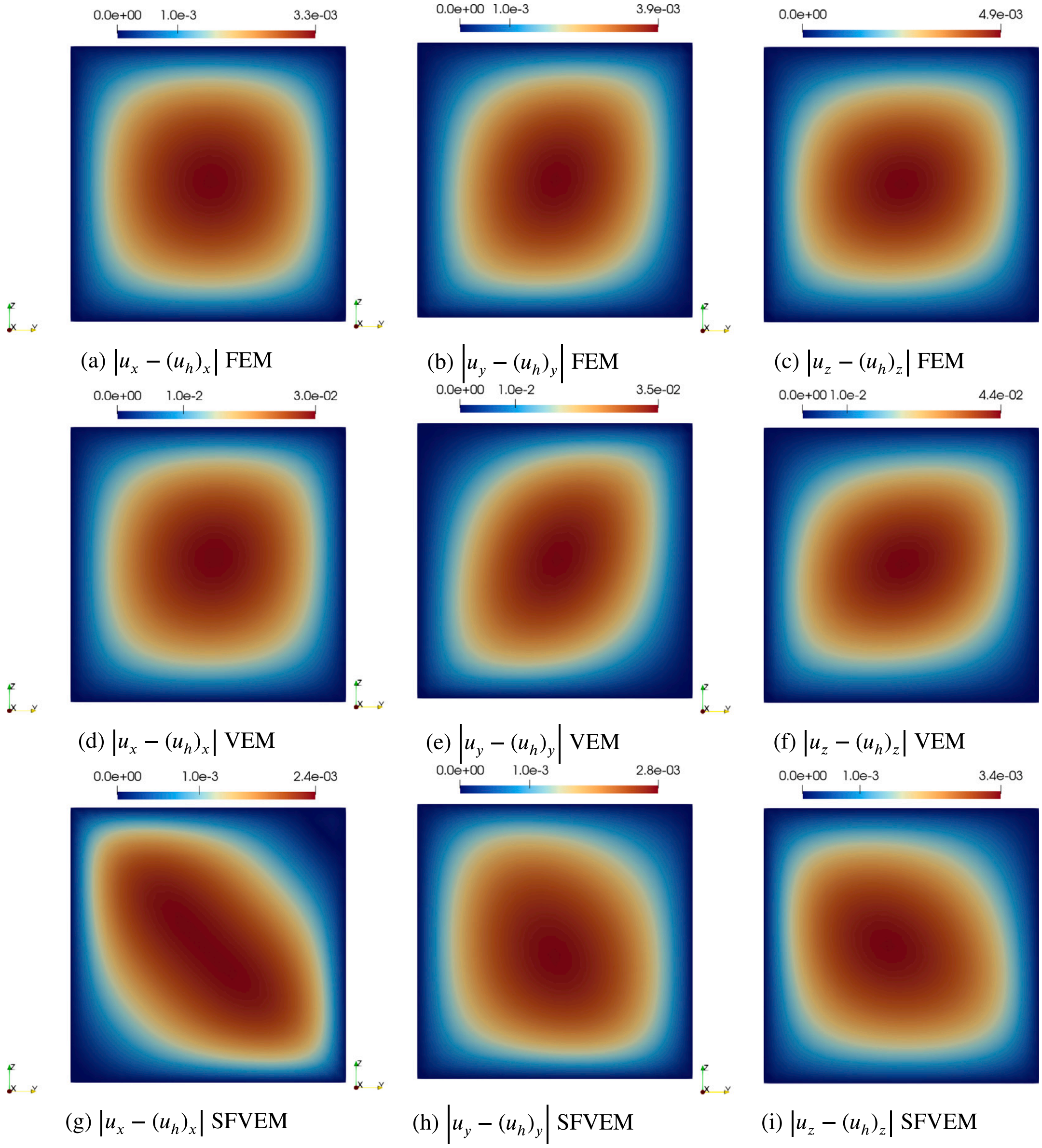
6.4. Nonlinear elasticity

In this section, we consider a nonlinear elastic problem. In particular, inspired by [35], we consider a benchmark problem, for which an analytical solution is available, given by: find $\boldsymbol{u} \in \mathbf{V}$ such that,

$$\int_\Omega \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{\varepsilon}(\boldsymbol{v}) = (\boldsymbol{f}, \boldsymbol{v})_\Omega \quad \forall \boldsymbol{v} \in \mathbf{V}, \tag{58}$$

where the constitutive law is given by

$$\boldsymbol{\sigma}(\boldsymbol{u}) := \hat{\mu}(\boldsymbol{\varepsilon}(\boldsymbol{u}))\, \boldsymbol{\varepsilon}(\boldsymbol{u}) \tag{59}$$



(a) $|u_x - (u_h)_x|$ FEM (b) $|u_y - (u_h)_y|$ FEM (c) $|u_z - (u_h)_z|$ FEM

(d) $|u_x - (u_h)_x|$ VEM (e) $|u_y - (u_h)_y|$ VEM (f) $|u_z - (u_h)_z|$ VEM

(g) $|u_x - (u_h)_x|$ SFVEM (h) $|u_y - (u_h)_y|$ SFVEM (i) $|u_z - (u_h)_z|$ SFVEM

Fig. 5. Distribution of the nodal error for the linear elastic case on the third refinement of the hexahedral mesh. We compute $|u_i - (u_h)_i|$ with $i = x, y, z$ on the cross-section defined by the plane through $[0.55, 0.5, 0.5]$ with normal vector $[1, 0, 0]$. Each row corresponds to a considered method, and each column corresponds to a ith component of $\boldsymbol{u}$.

and the scalar nonlinear function $\hat{\mu}$ is given by

$$\hat{\mu}(\boldsymbol{\varepsilon}(\boldsymbol{u})) := 3\left(1 + \|\boldsymbol{\varepsilon}(\boldsymbol{u})\|_2^2\right) \cdot 10^4$$

with $\|\boldsymbol{\varepsilon}\|_2^2 := \sum_{i,j} |\varepsilon_{ij}|^2$.

The discrete variational formulation applying the proposed stabilization-free VEM method reads as: find $\boldsymbol{u}_h \in \boldsymbol{\mathcal{V}}_{h,1}$ such that

$$\sum_{E \in \mathcal{M}_h} \int_E \hat{\mu}(\boldsymbol{\varepsilon}_h^E(\boldsymbol{u}_h))\, \boldsymbol{\varepsilon}_h^E(\boldsymbol{u}_h) : \boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h) = \sum_{E \in \mathcal{M}_h} \left(\boldsymbol{f}, \Pi_1^{0,E} \boldsymbol{v}_h\right)_E \quad \forall \boldsymbol{v}_h \in \boldsymbol{\mathcal{V}}_{h,1}, \tag{60}$$

where $\boldsymbol{\varepsilon}_h^E(\boldsymbol{u})$ is given in (21).

On the other hand, the standard VEM discrete formulation reads as: find $\boldsymbol{u}_h \in \boldsymbol{\mathcal{V}}_{h,1}$ such that

$$\sum_{E \in \mathcal{M}_h} \int_E \hat{\mu}(\boldsymbol{\varepsilon}_h^E(\boldsymbol{u}_h))\, \boldsymbol{\varepsilon}_h^E(\boldsymbol{u}_h) : \boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h) + S^E\left(\boldsymbol{u}_h, \boldsymbol{v}_h\right) = \sum_{E \in \mathcal{M}_h} \left(\boldsymbol{f}, \Pi_1^{0,E} \boldsymbol{v}_h\right)_E \quad \forall \boldsymbol{v}_h \in \boldsymbol{\mathcal{V}}_{h,1}, \tag{61}$$

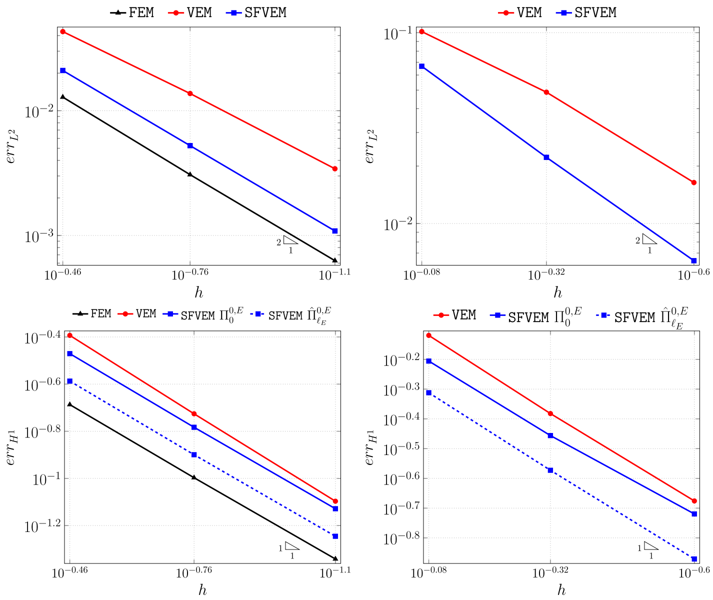


Fig. 6. Convergence plots for the Case 1 of the analyzed nonlinear elastic tests. Results on hexahedral meshes (left) and Voronoi meshes $\mathcal{D}_{\text{voro-Poisson}}$ (right). In the H^1-error plots, the solid line refers to the related measure in (54), the dashed line to the measure (57).

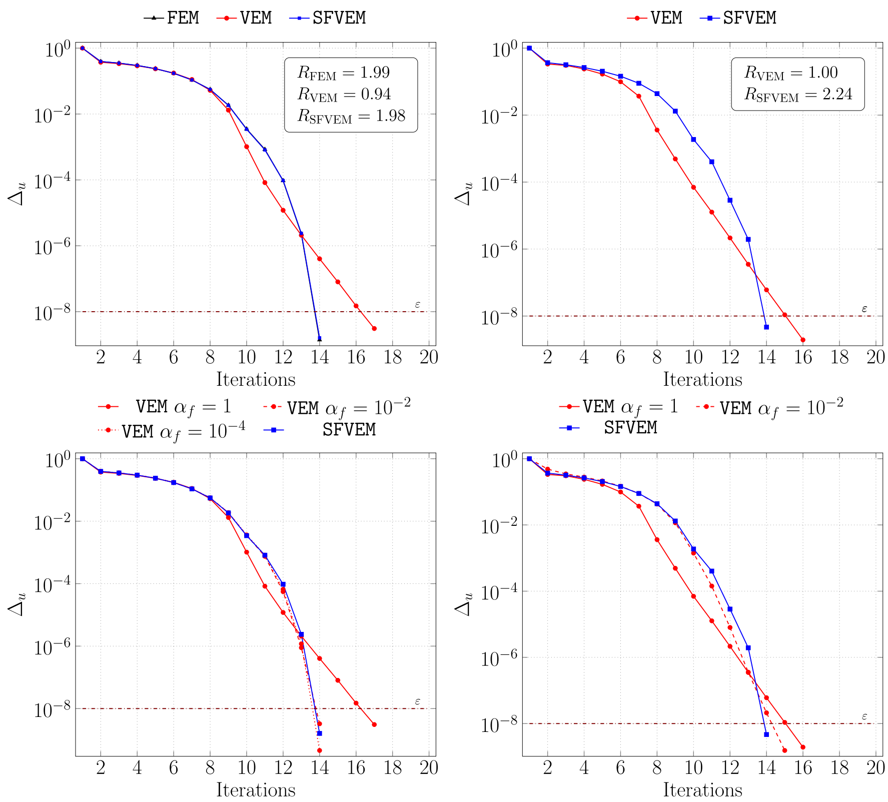


Fig. 7. Behavior of the quantity Δ_u in Case 1 of the analyzed nonlinear elastic tests. Newton tolerance is fixed to $\varepsilon = 10^{-8}$ (dashed horizontal red line). Each column corresponds to a different family of meshes. Results on hexahedral meshes (left) and Voronoi meshes $\mathcal{D}_{\text{voro-Poisson}}$ (right). Top: comparison for all the methods and rate of convergence. Bottom: SFVEM compared with VEM with different values of α_f.

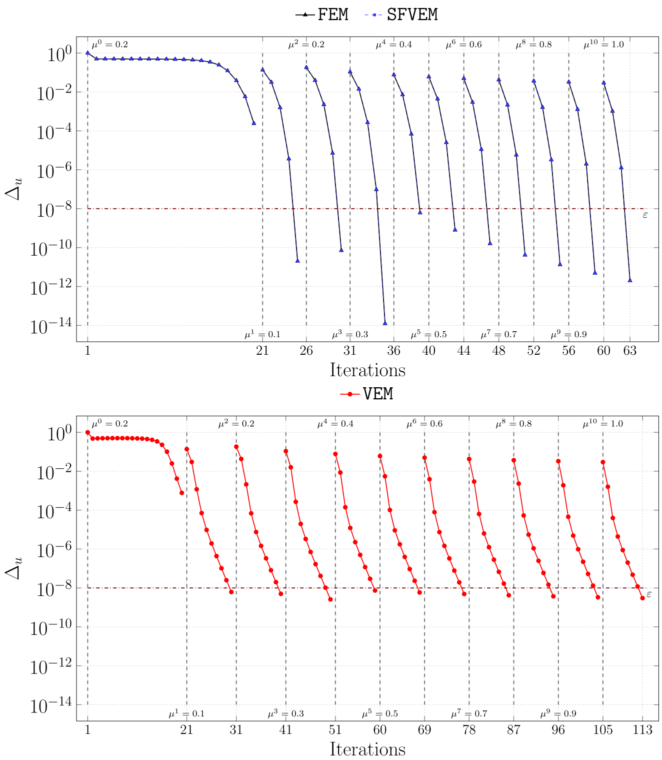


Fig. 8. Behavior of the quantity Δ_u in Case 2 of the analyzed nonlinear elastic tests. Newton tolerance is fixed to $\varepsilon = 10^{-8}$ (dashed horizontal red line). An adaptive loading-step procedure with an initial value $N = 5$ is used. The starting iteration of each loading step is marked by dashed black vertical lines, indicating the corresponding load increment μ^i. Top: FEM and SFVEM method are used. Bottom: VEM method is used.

where $\boldsymbol{\varepsilon}_h^E(\boldsymbol{u})$ is given in (55) and the stabilization term is the standard *dofi-dofi* defined in (56) with the problem-scaling parameter ς^E set, as suggested in [35], as

$$\varsigma^E(\boldsymbol{v}_h) = \alpha_f \left|\left|\left| \frac{\partial \hat{\mu}(\boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h))\boldsymbol{\varepsilon}_h^E(\boldsymbol{v}_h)}{\partial \nabla \boldsymbol{u}} \right|\right|\right| , \tag{62}$$

where $|||\cdot|||$ is the norm of the fourth order tensor space defined by the maximum of the absolute values of all the entries, and α_f is real a positive tuning parameter. To linearize the discrete problems introduced above, we employ the Newton–Raphson method, initialized with the zero solution $\boldsymbol{u}^0(x, y, z) = \boldsymbol{0}$ and a tolerance set to $\varepsilon = 10^{-8}$.

As in the referenced work, we consider two analytical solutions in our analysis:

$$\text{Case 1:} \quad \boldsymbol{u} = [u, u, u]^T, \quad u(x, y, z) := 64\, xyz(1-x)(1-y)(1-z) + 1.1;$$
$$\text{Case 2:} \quad \boldsymbol{u} = [u, u, u]^T, \quad u(x, y, z) := 640\, xyz(1-x)(1-y)(1-z) + 1.1.$$

We observe that in Case 1, we adopt the same solution as in the linear problem considered in the previous section, and in Case 2, the magnitude of the load is increased to simulate larger deformations.

We solve Case 1 on the first three meshes of the families considered in the previous test. Fig. 6 illustrates the behavior of the SFVEM errors in L^2 and H^1 norms, compared with FEM and VEM on hexahedral meshes, and with VEM only on Voronoi meshes. Tetrahedral meshes are omitted, as all three methods exhibit the expected similar behavior. As in the linear case, the expected convergence rates are achieved by all the methods.

We report in Fig. 7 the Newton–Raphson discrete solution convergence for Case 1 at each iteration, using the finest hexahedral and Voronoi meshes in the dataset. Specifically, given the index n corresponding to the nth iteration of the Newton–Raphson method, let $\boldsymbol{u}^n$ denote the solution at the nth iteration. We measure the variation of the discrete solution at the $(n+1)$th step of the method as

$$\Delta_u := \frac{\left\| \Pi_1^{0,E} \left(\boldsymbol{u}^{n+1} - \boldsymbol{u}^n\right)\right\|_{\mathrm{L}^2(\Omega)}}{\left\| \Pi_1^{0,E} \boldsymbol{u}^{n+1} \right\|_{\mathrm{L}^2(\Omega)}} . \tag{63}$$

We remark that the above measure simplifies to the standard L^2 norm for the finite element functions. Moreover, we evaluate the standard Newton–Raphson residual based on the norm of the right-hand side of the system. Since these measure yields results fully consistent with those obtained using the previously defined Δ_u measure, the corresponding results are not reported. We report, in the upper right part of the plots in Fig. 7, a table containing the convergence rates measured at the last Newton iteration for each method considered, namely R_{FEM}, R_{VEM}, and R_{SFVEM}. From these values, we observe that the proposed SFVEM exhibits the expected quadratic convergence of the Newton method, matching the behavior observed for the FEM solution on the hexahedral mesh, while behavior of the standard VEM (with the stabilization tuning parameter α_f set equal to 1) is limited to linear convergence. The slower convergence of the standard VEM method is due to the presence of the stabilization term in the Newton–Raphson matrices, which strongly affects the Jacobian direction.

To further investigate this effect, we tune the parameter $\alpha_f \in \{1, 10^{-2}, 10^{-4}\}$, that scales the stabilization term, thereby reducing its magnitude. In the bottom part of Fig. 7, we show the Δ_u convergence for varying values of α_f. The plots indicate that the Newton convergence rate of VEM improves as the stabilization magnitude decreases. In particular, for the hexahedral mesh and $\alpha_f = 10^{-4}$, the speed rate of VEM becomes close to quadratic. However, we emphasize that the parameter α_f is non-physical and strongly depends on both the problem data and the discretization. Indeed, the same value $\alpha_f = 10^{-4}$ cannot be used for the Voronoi mesh and is therefore not reported in the plots, as we observe that it leads to a loss of convergence of the Newton algorithm to the exact solution.

Finally, we consider the Newton–Raphson method applied to the Case 2. In Fig. 8, we present the Δ_u convergence for Case 2 on the finest hexahedral mesh. Due to the increased magnitude of the forcing load $\boldsymbol{f}$, a load-stepping strategy is employed in the numerical implementation, in line with the reference work [35]. Specifically, we apply an usual incremental loading procedure for the solution of the nonlinear discrete problem: given a positive integer N, let the partial loadings $\boldsymbol{f}^i = \mu^i \boldsymbol{f}^i$, with $\mu^i = i/N$ for all $i = 1, \dots, N$.

First, we set $N = 5$. At each loading step, up to 20 Newton iterations are performed. When convergence of the Newton scheme is not achieved, the next load increment μ^i is reduced by a factor of 0.5, effectively doubling the number of the remaining loading steps. This standard strategy is used to improve the convergence properties of the Newton–Raphson method, by constructing an initial guess that lies sufficiently close to the basin of attraction.

From the two plots in Fig. 8, we observe that only one reduction of a factor 0.5 of the increment is sufficient to achieve convergence for all the discrete methods. Even applying the loading step procedure, we observe that the proposed SFVEM method attains the expected quadratic convergence, similarly to the simulation employing FEM, whereas the standard VEM (with the stabilization tuning parameter α_f set equal to 1) exhibits still linear convergence, as in the previous case. Hence, the behavior of the standard VEM method is strongly influenced by the presence of the stabilization term, even applying the loading step procedure. Consequently, the total number of Newton's iterations required by the standard VEM approach is approximately twice the one needed by the proposed SFVEM method, resulting in a significantly reduced computational cost and improved overall computational efficiency. Finally, we recall that the computation of the local projector for each polyhedron – and, consequently, the execution of Algorithm 1 – is required only once (see Remark 4).

Conclusions

In this work, we proposed a novel virtual element method for three-dimensional second-order elliptic equations and elasticity problems that does not require the use of an arbitrary non polynomial stabilization term. The proposed stabilization-free scheme is based on the definition of a local high-order polynomial projection of the gradient of the basis functions. We identified a sufficient condition for choosing the degree of the polynomial projection and we proved that the proposed method is well-posed if that condition is satisfied. Moreover, we proposed a local QR-based algorithm that provides the local polynomial projection degree ensuring stability, while discussing the computability of the projector. Numerical experiments on general polyhedral meshes show the local stability and robustness of the proposed scheme. In all tests, low polynomial projection degrees are achieved, even for polyhedra with a large number of vertices. Finally, we apply the proposed method to a nonlinear elasticity problem, showing that it achieves quadratic convergence of the Newton scheme independently of the problem data. We remark that the proposed method results to be more robust with respect to a standard VEM formulation for the solution of nonlinear problems, as it does not require the tuning of any non-physical parameter, and significantly reduces the computational cost and the overall efficiency.

CRediT authorship contribution statement

Stefano Berrone: Supervision, Resources, Project administration, Methodology, Funding acquisition, Conceptualization. **Andrea Borio:** Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Francesca Marcon:** Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Fabio Vicini:** Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

The four authors are members of the Gruppo Nazionale Calcolo Scientifico (GNCS) at Istituto Nazionale di Alta Matematica (INdAM). The authors kindly acknowledge financial support by the European Union through Next Generation EU, M4C2, PRIN 2022 PNRR project (CUP: E53D23017950001) and by INdAM-GNCS Projects 2025 (CUP: E53C24001950001) and 2026 (CUP: E53C25002010001).

Data availability

Data will be made available on request.

References

[1] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L.D. Marini, A. Russo, Basic principles of virtual element methods, Math. Models Methods Appl. Sci. 23 (01) (2013) 199–214, http://dx.doi.org/10.1142/S0218202512500492.
[2] A. D'Altri, S. de Miranda, L. Patruno, E. Sacco, An enhanced VEM formulation for plane elasticity, Comput. Methods Appl. Mech. Engrg. 376 (2021) 113663, http://dx.doi.org/10.1016/j.cma.2020.113663.
[3] S. Berrone, A. Borio, F. Marcon, G. Teora, A first-order stabilization-free virtual element method, Appl. Math. Lett. 142 (2023) 108641, http://dx.doi.org/10.1016/j.aml.2023.108641.
[4] A. Lamperti, M. Cremonesi, U. Perego, C. Lovadina, A. Russo, A Hu-Washizu variational approach to self-stabilized Virtual Elements: 2D linear elastostatics, Comput. Mech. 71 (2023) 935–955, http://dx.doi.org/10.1007/s00466-023-02282-2.
[5] S. Berrone, A. Borio, F. Marcon, Lowest order stabilization free virtual element method for the 2D Poisson equation, Comput. Math. Appl. 177 (2025) 78–99, http://dx.doi.org/10.1016/j.camwa.2024.11.017.
[6] A. Borio, C. Lovadina, F. Marcon, M. Visinoni, A lowest order stabilization-free mixed virtual element method, Comput. Math. Appl. 160 (2024) 161–170, http://dx.doi.org/10.1016/j.camwa.2024.02.024.
[7] S. Berrone, A. Borio, F. Marcon, Comparison of standard and stabilization free virtual elements on anisotropic elliptic problems, Appl. Math. Lett. 129 (2022) 107971, http://dx.doi.org/10.1016/j.aml.2022.107971.
[8] A. Borio, M. Busetto, F. Marcon, SUPG-stabilized stabilization-free VEM: a numerical investigation, Math. Eng. 6 (1) (2024) 173–191, http://dx.doi.org/10.3934/mine.2024008.
[9] A. Chen, N. Sukumar, Stabilization-free virtual element method for plane elasticity, Comput. Math. Appl. 138 (2023) 88–105, http://dx.doi.org/10.1016/j.camwa.2023.03.002.
[10] B.-B. Xu, F. Peng, P. Wriggers, Stabilization-free virtual element method for finite strain applications, Comput. Methods Appl. Mech. Engrg. 417 (2023) 116555, http://dx.doi.org/10.1016/j.cma.2023.116555.
[11] B.-B. Xu, Y. Wang, P. Wriggers, Stabilization-free virtual element method for 2D elastoplastic problems, Internat. J. Numer. Methods Engrg. (2024) e7490, http://dx.doi.org/10.1002/nme.7490.
[12] S. Berrone, A. Borio, D. Fassino, F. Marcon, A residual a posteriori error estimate for the Stabilization-free Virtual Element Method, J. Comput. Phys. 553 (2026) 114704, http://dx.doi.org/10.1016/j.jcp.2026.114704.
[13] B.-B. Xu, P. Wriggers, 3D stabilization-free virtual element method for linear elastic analysis, Comput. Methods Appl. Mech. Engrg. 421 (2024) 116826, http://dx.doi.org/10.1016/j.cma.2024.116826.
[14] B.-B. Xu, F. Peng, P. Wriggers, Stabilization-free virtual element method for 3D hyperelastic problems, Comput. Mech. 75 (6) (2025) 1687–1701, http://dx.doi.org/10.1007/s00466-024-02501-4.
[15] D. Sun, E. Pescialli, Q. Li, M. Cremonesi, C. Lovadina, U. Perego, A. Russo, Deltahedral self-stabilized virtual elements for 3D linear elastostatics problems, Comput. Mech. 76 (3) (2025) 727–743, http://dx.doi.org/10.1007/s00466-025-02622-4.
[16] F. Credali, S. Bertoluzza, D. Prada, Reduced basis stabilization and post-processing for the virtual element method, Comput. Methods Appl. Mech. Engrg. 420 (2024) 116693, http://dx.doi.org/10.1016/j.cma.2023.116693.
[17] L. Beirão da Veiga, C. Canuto, R.H. Nochetto, G. Vacca, M. Verani, Adaptive VEM: Stabilization-free a posteriori error analysis and contraction property, SIAM J. Numer. Anal. 61 (2) (2023) 457–494, http://dx.doi.org/10.1137/21M1458740.
[18] C. Canuto, D. Fassino, Higher-order adaptive virtual element methods with contraction properties, Math. Eng. 5 (6) (2023) 1–33, http://dx.doi.org/10.3934/mine.2023101.
[19] S. Berrone, D. Fassino, F. Vicini, 3D Adaptive VEM with Stabilization-Free a posteriori error bounds, J. Sci. Comput. 103 (1) (2025) 35, http://dx.doi.org/10.1007/s10915-025-02852-x.
[20] S. Berrone, M. Pintore, G. Teora, The neural approximated virtual element method for elasticity problems, Finite Elem. Anal. Des. 252 (2025) 104467, http://dx.doi.org/10.1016/j.finel.2025.104467.
[21] D.A. Di Pietro, J. Droniou, G. Manzini, Discontinuous Skeletal Gradient Discretisation methods on polytopal meshes, J. Comput. Phys. 355 (2018) 397–425, http://dx.doi.org/10.1016/j.jcp.2017.11.018.
[22] A. Borio, K.L. Cascavita, M. Cicuttin, F. Marcon, Towards stabilization-free Hybrid High-Order methods for elliptic problems, J. Sci. Comput. 104 (2025) 83, http://dx.doi.org/10.1007/s10915-025-02985-z.
[23] S. Berrone, A. Borio, F. Marcon, A stabilization-free virtual element method based on divergence-free projections, Comput. Methods Appl. Mech. Engrg. 424 (2024) 116885, http://dx.doi.org/10.1016/j.cma.2024.116885.
[24] S. Berrone, A. Borio, D. Fassino, F. Marcon, Stabilization-free virtual element method for 2D second order elliptic equations, Comput. Methods Appl. Mech. Engrg. 438 (2025) http://dx.doi.org/10.1016/j.cma.2025.117839, Paper No. 117839, 24, ISSN: 0045-7825, 1879-2138.
[25] A. Ern, J.-L. Guermond, Finite Elements II: Galerkin Approximation, Elliptic and Mixed PDEs, Springer International Publishing, 2021, pp. 253–268, http://dx.doi.org/10.1007/978-3-030-56923-5_42.
[26] B. Ahmad, A. Alsaedi, F. Brezzi, L.D. Marini, A. Russo, Equivalent projectors for virtual element methods, Comput. Math. Appl. 66 (2013) 376–391, http://dx.doi.org/10.1016/j.camwa.2013.05.015.
[27] L. Beirão da Veiga, F. Brezzi, L.D. Marini, A. Russo, Virtual element methods for general second order elliptic problems on polygonal meshes, Math. Models Methods Appl. Sci. 26 (04) (2015) 729–750, http://dx.doi.org/10.1142/S0218202516500160.
[28] S.C. Brenner, L.-Y. Sung, Virtual element methods on meshes with small edges or faces, Math. Models Methods Appl. Sci. 28 (07) (2018) 1291–1336, http://dx.doi.org/10.1142/S0218202518500355.
[29] L. Beirão da Veiga, F. Brezzi, F. Dassi, L.D. Marini, A. Russo, A family of three-dimensional virtual elements with applications to magnetostatics, SIAM J. Numer. Anal. 56 (5) (2018) 2940–2962, http://dx.doi.org/10.1137/18M1169886.

[30] L. Beirão da Veiga, F. Brezzi, L.D. Marini, A. Russo, $H(div)$ and $H(\mathbf{curl})$-conforming virtual element methods, Numer. Math. 133 (2) (2016) 303–332, http://dx.doi.org/10.1007/s00211-015-0746-1.

[31] L. Beirão da Veiga, F. Brezzi, L.D. Marini, A. Russo, The Hitchhiker's guide to the virtual element method, Math. Models Methods Appl. Sci. 24 (08) (2014) 1541–1573, http://dx.doi.org/10.1142/S021820251440003X.

[32] A. Sky, I. Muench, P. Neff, On $[H^1]^3 \times 3$, $[H(curl)]^3$ and $H(symCurl)$ finite elements for matrix-valued Curl problems, J. Engrg. Math. 136 (2022) 5, http://dx.doi.org/10.1007/s10665-022-10238-3.

[33] L. Beirão da Veiga, F. Brezzi, L.D. Marini, Virtual elements for linear elasticity problems, SIAM J. Numer. Anal. 51 (2) (2013) 794–812, http://dx.doi.org/10.1137/120874746.

[34] F. Dassi, S. Scacchi, Parallel solvers for virtual element discretizations of elliptic equations in mixed form, Comput. Math. Appl. 79 (7) (2020) 1972–1989, http://dx.doi.org/10.1016/j.camwa.2019.07.027, Advanced Computational methods for PDEs.

[35] L. Beirão da Veiga, C. Lovadina, D. Mora, A virtual element method for elastic and inelastic problems on polytope meshes, Comput. Methods Appl. Mech. Engrg. 295 (2015) 327–346, http://dx.doi.org/10.1016/j.cma.2015.07.013.

[36] S. Berrone, A. Borio, G. Teora, F. Vicini, POLYDIM: A C++ library for polytopal discretization methods, Comput. Phys. Comm. 320 (2026) 109937, http://dx.doi.org/10.1016/j.cpc.2025.109937.

[37] T. Sorgente, S. Biasotti, G. Manzini, M. Spagnuolo, Polyhedral mesh quality indicator for the virtual element method, Comput. Math. Appl. 114 (2022) 151–160, http://dx.doi.org/10.1016/j.camwa.2022.03.042.

[38] A.L. Gain, C. Talischi, G.H. Paulino, On the Virtual Element Method for three-dimensional linear elasticity problems on arbitrary polyhedral meshes, Comput. Methods Appl. Mech. Engrg. 282 (2014) 132–160, http://dx.doi.org/10.1016/j.cma.2014.05.005.

[39] H. Si, TetGen, a Delaunay-Based Quality Tetrahedral Mesh Generator, ACM Trans. Math. Software 41 (2) (2015) http://dx.doi.org/10.1145/2629697.