Orthogonal polynomial bases in the mixed virtual element method

Received: 28 April 2023 | Revised: 8 May 2024 | Accepted: 7 August 2024

DOI: 10.1002/num.23144

RESEARCH ARTICLE

WILEY

# Orthogonal polynomial bases in the mixed virtual element method

**Stefano Berrone** | **Stefano Scialò** | **Gioana Teora**

Dipartimento di Scienze Matematiche "G. L. Lagrange", Politecnico di Torino, Turin, Italy

**Correspondence**
Gioana Teora, Dipartimento di Scienze Matematiche "G. L. Lagrange", Politecnico di Torino, Turin, Italy.
Email: gioana.teora@polito.it

**Funding information**
Ministero dell'Istruzione, dell'Università e della Ricerca; Ministero dell'Università e della Ricerca; Gruppo Nazionale per il Calcolo Scientifico.

## Abstract

The use of orthonormal polynomial bases has been found to be efficient in preventing ill-conditioning of the system matrix in the primal formulation of Virtual Element Methods (VEM) for high values of polynomial degree and in presence of badly-shaped polygons. However, we show that using the natural extension of a orthogonal polynomial basis built for the primal formulation is not sufficient to cure ill-conditioning in the mixed case. Thus, in the present work, we introduce an orthogonal vector-polynomial basis which is built ad hoc for being used in the mixed formulation of VEM and which leads to very high-quality solution in each tested case. Furthermore, a numerical experiment related to simulations in Discrete Fracture Networks (DFN), which are often characterised by very badly-shaped elements, is proposed to validate our procedures.

**KEYWORDS**

ill-conditioning, mixed VEM, orthogonal polynomial basis

## 1 | INTRODUCTION

The Mixed Virtual Element Methods were introduced originally in [11] for the Poisson problem in the two-dimensional case and then were extended to more general elliptic equations in [6]. In the Mixed Virtual Element Space, two discrete spaces are introduced for approximating the pressure variable and the velocity field, respectively. The first space is a scalar-polynomial space, while a vector-polynomial basis is required to build the local projection matrices and for defining the internal degrees of freedom needed to obtain an approximation of the velocity field. It was observed that using the classical choice

of scaled monomials in the definition of internal degrees of freedom in the primal Virtual Element construction [3, 4, 7, 11], the system matrix could become ill-conditioned in presence of *badly-shaped* polygons [9] (e.g., collapsing edges and bulks) and for high values of local polynomial degree [15].

The present work aims at defining new polynomial bases for the mixed VEM construction yielding well-conditioned local projection matrices also in presence of badly-shaped elements. More precisely, we propose two different approaches for building a vector-polynomial basis, which we briefly called "Partial" and "Ortho", respectively. The first one is the natural extension to the mixed case of the approach presented in [15] for the primal VEM, which allows us to build a vector-polynomial basis that is only partially $L^2$-orthonormalized. We show that the use of such basis is not sufficient to cure the ill-conditioning of the system matrix related to the mixed formulation of VEM in all circumstances [1], even if, in the primal setting, it reveals to be the best alternative. Thus, we introduce the Ortho approach which aims to orthogonalize the gradients of a proper scalar-polynomial basis in order to obtain a full orthonormal vector-polynomial basis. We show that this approach leads to the best local and global performances, throughout different numerical experiments characterised by challenging geometries.

The outline of the present paper is the following. We define the model problem in Section 2 and its mixed VEM approximation in Section 3. In Section 4, we describe how to build the new polynomial bases, while in Section 5 we show an efficient implementation of the method, totally matrix-based. Finally, in Section 6, we perform some numerical experiments that show the advantages of using the new polynomial bases.

Let us introduce some notations used throughout the paper. Given $k \in \mathbb{N}$, we use $(\cdot,\cdot)_{k,\sigma}$ and $\|\cdot\|_{k,\sigma}$ to indicate the inner product and the norm in the Sobolev space $H^k(\sigma)$ on some open subset $\sigma \subset \mathbb{R}^2$, respectively. Furthermore, if $\boldsymbol{v} = \left[v_1, v_2\right]^T$ and $\boldsymbol{u} = \left[u_1, u_2\right]^T$ are vectors in $L^2(\sigma) \times L^2(\sigma)$, we define

$$(\boldsymbol{v}, \boldsymbol{u})_{0,\sigma} = \int_\sigma (v_1 u_1 + v_2 u_2), \quad \|\boldsymbol{v}\|_{0,\sigma} = \sqrt{(\boldsymbol{v}, \boldsymbol{v})_{0,\sigma}}. \tag{1}$$

Let $\Omega \subset \mathbb{R}^2$ be a bounded convex polygonal domain with boundary $\Gamma$ and let $\boldsymbol{n}_\Gamma$ be the outward unit normal vector to $\Gamma$, then we define the functional spaces

$$H(\mathrm{div};\Omega) = \left\{\boldsymbol{v} \in L^2(\Omega) \times L^2(\Omega) \,:\, \nabla \cdot \boldsymbol{v} \in L^2(\Omega)\right\}, \tag{2}$$

$$H_{0,\Gamma_N}(\mathrm{div};\Omega) = \{\boldsymbol{v} \in H(\mathrm{div},\Omega) \,:\, \boldsymbol{v} \cdot \boldsymbol{n}_\Gamma = 0 \text{ on } \Gamma_N \subseteq \Gamma\}, \tag{3}$$

$$H(\mathrm{rot}\,;\Omega) = \{\boldsymbol{v} \in L^2(\Omega) \times L^2(\Omega) \,:\, \mathrm{rot}\,\boldsymbol{v} \in L^2(\Omega)\}. \tag{4}$$

Furthermore, let $H^{-\frac{1}{2}}(\Gamma)$ be the dual space of the Sobolev space $H^{\frac{1}{2}}(\Gamma)$, the symbol $\langle\cdot,\cdot\rangle_{\pm\frac{1}{2},\Gamma}$ denotes the duality pairing between $H^{-\frac{1}{2}}(\Gamma)$ and $H^{\frac{1}{2}}(\Gamma)$.

## 2 | THE CONTINUOUS PROBLEM AND THE MIXED VARIATIONAL FORMULATION

Let $\boldsymbol{\kappa}$ be a symmetric uniformly positive definite tensor over $\Omega$, $\gamma$ a sufficiently smooth function $\Omega \to \mathbb{R}$ and $\boldsymbol{b}$ a smooth vector valued function $\Omega \to \mathbb{R}^2$. We consider the following problem:

$$\begin{cases} \nabla \cdot (-\boldsymbol{\kappa}\nabla p + \boldsymbol{b}p) + \gamma p = f & \text{in } \Omega, \\ p = g_D & \text{on } \Gamma_D, \\ (-\boldsymbol{\kappa}\nabla p + \boldsymbol{b}p) \cdot \boldsymbol{n}_{\Gamma_N} = g_N & \text{on } \Gamma_N, \end{cases} \tag{5}$$

where $\Gamma_D$ and $\Gamma_N$ are the Dirichlet and the Neumann boundary, respectively, such that $\Gamma_D \cup \Gamma_N = \Gamma$ and $|\Gamma_D \cap \Gamma_N| = 0$.

In order to introduce the mixed variational formulation, we define

$$\boldsymbol{K} = \boldsymbol{\kappa}^{-1}, \quad \boldsymbol{\beta} = \boldsymbol{K}\boldsymbol{b}, \tag{6}$$

and we re-write problem (5) as

$$\begin{cases} \boldsymbol{K}\boldsymbol{u} = -\nabla p + \boldsymbol{\beta} p & \text{in } \Omega, \\ \nabla \cdot \boldsymbol{u} + \gamma p = f & \text{in } \Omega, \\ p = g_D & \text{on } \Gamma_D, \\ \boldsymbol{u} \cdot \boldsymbol{n}_{\Gamma_N} = g_N & \text{on } \Gamma_N. \end{cases} \tag{7}$$

Thus, the mixed variational formulation of (5) reads:

Find $\boldsymbol{u} = \boldsymbol{u}_0 + \boldsymbol{u}_N$, with $\boldsymbol{u}_0 \in \boldsymbol{V} = H_{0,\Gamma_N}(\mathrm{div}; \Omega)$, and $p \in Q = L^2(\Omega)$ such that

$$\begin{cases} (\boldsymbol{K}\boldsymbol{u}, \boldsymbol{v})_{0,\Omega} - (p, \nabla \cdot \boldsymbol{v})_{0,\Omega} - (\boldsymbol{\beta} p, \boldsymbol{v})_{0,\Omega} = -\langle g_D, \boldsymbol{v} \cdot \boldsymbol{n}_{\Gamma_D} \rangle_{\frac{1}{2},\Gamma_D} & \forall \boldsymbol{v} \in \boldsymbol{V} \\ (\nabla \cdot \boldsymbol{u}, q)_{0,\Omega} + (\gamma p, q)_{0,\Omega} = (f, q)_{0,\Omega} & \forall q \in Q \end{cases} \tag{8}$$

where $\boldsymbol{u}_N \in H(\mathrm{div}; \Omega)$ is a chosen function that satisfies $\boldsymbol{u}_N \cdot \boldsymbol{n}_{\Gamma_N} = g_N$ on $\Gamma_N$.

## 3 | THE MIXED VIRTUAL ELEMENT METHOD

Let $\mathcal{T}_h$ be a decomposition of $\Omega$ into star-shaped polygons $E$. We will denote by $\boldsymbol{x}_E$, $h_E$ and $\mathcal{E}_{h,E}$ the centroid, the diameter and the set of edges of $E$, respectively. We further set $N^{E,e} = \#\mathcal{E}_{h,E}$, and, as usual, we fix $h = \max_{E \in \mathcal{T}_h} h_E$.

Moreover, $\mathbb{P}_k(E)$ is the set of all polynomials defined on $E$ of degree less or equal to $k \geq 0$ and $n_k = \dim \mathbb{P}_k(E) = \frac{(k+1)(k+2)}{2}$. For the ease of the notation, we fix $\mathbb{P}_{-1} = \{0\}$ and $n_{-1} = 0$ and we introduce the natural function $\ell : \mathbb{N}^2 \to \mathbb{N}$ which associates

$$(0,0) \mapsto 1, \quad (1,0) \to 2, \quad (0,1) \mapsto 3, \quad (2,0) \mapsto 4, \quad (1,1) \to 5, \ \ldots \tag{9}$$

A classical choice of the basis for $\mathbb{P}_k(E)$ that can be found in virtual element literature (see [3, 4, 7]) is the set of the scaled monomials, which can be defined as

$$\mathcal{M}_k(E) = \left\{ \mathbb{m}_\alpha^k = \left( \frac{\boldsymbol{x} - \boldsymbol{x}_E}{h_E} \right)^{\boldsymbol{\alpha}}, \forall \boldsymbol{\alpha} = \ell(\alpha) \in \mathbb{N}^2 \text{ s.t. } \alpha = 1, \ldots, n_k \right\}, \tag{10}$$

where the function $\ell$ is defined in (9).

As in [6], we introduce the (vector) polynomial space

$$\mathcal{G}_k^\nabla(E) = \nabla \mathbb{P}_{k+1}(E) = \left\{ \boldsymbol{g}_\alpha^{\nabla,k} \right\}_{\alpha=1}^{n_k^\nabla} \tag{11}$$

and its complement $\mathcal{G}_k^\perp(E) = \left\{ \boldsymbol{g}_\alpha^{\perp,k} \right\}_{\alpha=1}^{n_k^\perp}$ in $\left[\mathbb{P}_k(E)\right]^2$, which satisfies

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

where $\bigoplus$ is the direct sum operator, and

$$\dim \left[\mathbb{P}_k(E)\right]^2 = 2n_k, \tag{13}$$

$$n_k^\nabla = \dim \mathcal{G}_k^\nabla(E) = n_k + (k+1), \tag{14}$$

$$n_k^{\perp} = \dim \mathcal{G}_k^{\perp}(E) = n_k - (k+1). \tag{15}$$

Now, following [6], for any integer $k \geq 0$, we define the local mixed virtual element space for the velocity variable $\boldsymbol{u}$ as

$$\begin{aligned} \boldsymbol{V}_{h,k}(E) = \{\boldsymbol{v}_h \in H(\mathrm{div};E) \cap H(\mathrm{rot};E) \text{ s.t. } \boldsymbol{v}_h \cdot \boldsymbol{n}_e \in \mathbb{P}_k(e)\ \forall e \in \mathcal{E}_{h,E}, \\ \nabla \cdot \boldsymbol{v}_h \in \mathbb{P}_k(E),\ \mathrm{rot}\,\boldsymbol{v}_h \in \mathbb{P}_{k-1}(E)\}. \end{aligned} \tag{16}$$

It is easy to see that $\left[\mathbb{P}_k(E)\right]^2 \subset \boldsymbol{V}_{h,k}(E)$.

The following set of local degrees of freedom is unisolvent for $\boldsymbol{V}_{h,k}(E)$ (see [5, 7]): given $\boldsymbol{v}_h \in \boldsymbol{V}_{h,k}(E)$,

- **Edge dofs**: chosen $k+1$ Gauss quadrature points $\boldsymbol{x}_i^{e,Q}$ internal on each edge $e \in \mathcal{E}_{h,E}$:

$$\mathrm{dof}_i^e(\boldsymbol{v}_h) = (\boldsymbol{v}_h \cdot \boldsymbol{n}_e)(\boldsymbol{x}_i^{e,Q}) \quad \forall i = 1, \dots, k+1. \tag{17}$$

  We note that this choice automatically ensures the continuity of the flux $\boldsymbol{v}_h \cdot \boldsymbol{n}$ across two adjacent elements.
- **Internal $\nabla$ dofs**:

$$\mathrm{dof}_\alpha^{\nabla}(\boldsymbol{v}_h) = \frac{1}{|E|}\int_E \boldsymbol{v}_h \cdot \boldsymbol{g}_\alpha^{\nabla,k-1} \quad \forall \alpha = 1, \dots, n_{k-1}^{\nabla}. \tag{18}$$

- **Internal $\perp$ dofs**:

$$\mathrm{dof}_\alpha^{\perp}(\boldsymbol{v}_h) = \frac{1}{|E|}\int_E \boldsymbol{v}_h \cdot \boldsymbol{g}_\alpha^{\perp,k} \quad \forall \alpha = 1, \dots, n_k^{\perp}. \tag{19}$$

Let it be $N_E^{\mathrm{dof}} = \dim \boldsymbol{V}_{h,k}(E) = N^{E,e}(k+1) + n_{k-1}^{\nabla} + n_k^{\perp}$, we denote henceforth the local Lagrangian mixed VE basis corresponding to the defined degrees of freedom:

$$\left\{\boldsymbol{\varphi}_i\right\}_{i=1}^{N_E^{\mathrm{dof}}} = \left\{\left\{\boldsymbol{\varphi}_i^e\right\}_{i=1}^{k+1}\right\}_{e\in\mathcal{E}_{h,E}} \cup \left\{\boldsymbol{\varphi}_i^{\nabla}\right\}_{i=1}^{n_{k-1}^{\nabla}} \cup \left\{\boldsymbol{\varphi}_i^{\perp}\right\}_{i=1}^{n_k^{\perp}}, \tag{20}$$

where the dofs numbering first counts the edge dofs, then the internal $\nabla$ dofs and lastly the internal $\perp$ dofs.

As in [6], we define the local mixed virtual element space $Q_{h,k}(E)$ for the pressure variable $p$ as the space of polynomials $\mathbb{P}_k(E)$, that is, $Q_{h,k}(E) = \mathbb{P}_k(E)$. In the next section, we will provide further details regarding the selection of the local basis functions for the local pressure space.

Finally, we define the global mixed virtual element spaces for both velocity and pressure variables as

$$\boldsymbol{V}_{h,k} = \{\boldsymbol{v}_h \in H_{0,\Gamma_N}(\mathrm{div};\Omega) \text{ s.t. } \boldsymbol{v}_{h|E} \in \boldsymbol{V}_{h,k}(E)\ \forall E \in \mathcal{T}_h\}, \tag{21}$$

$$Q_{h,k} = \{q_h \in L^2(\Omega) \text{ s.t. } q_{h|E} \in Q_{h,k}(E)\ \forall E \in \mathcal{T}_h\}. \tag{22}$$

### 3.1 | The discrete mixed variational formulation

The $L^2(E)$-projection operator $\Pi_k^0 : \boldsymbol{V}_{h,k} \to \left[\mathbb{P}_k(\mathcal{T}_h)\right]^2$ is locally defined as

$$\left(\Pi_k^0 \boldsymbol{v}_h, \boldsymbol{p}_k\right)_{0,E} = \left(\boldsymbol{v}_h, \boldsymbol{p}_k\right)_{0,E} \forall \boldsymbol{p}_k \in \left[\mathbb{P}_k(E)\right]^2 \text{ and } \forall E \in \mathcal{T}_h. \tag{23}$$

and, as shown in [5], the projection $\Pi_k^0 \boldsymbol{v}_h$ of a virtual function $\boldsymbol{v}_h \in \boldsymbol{V}_{h,k}$ can be explicitly computed from the knowledge of its degrees of freedom (17)–(19).

Now, the local discrete counterpart of the continuous bilinear form

$$a(\boldsymbol{u},\boldsymbol{v}) = (\boldsymbol{K}\boldsymbol{u},\boldsymbol{v})_{0,\Omega}, \quad \forall \boldsymbol{u},\ \boldsymbol{v} \in \boldsymbol{V}, \tag{24}$$

reads

$$a_h(\boldsymbol{u}_h,\boldsymbol{v}_h) = \sum_{E\in\mathcal{T}_h} a_h^E(\boldsymbol{u}_h,\boldsymbol{v}_h) \tag{25}$$

$$= \sum_{E\in\mathcal{T}_h} \Big[ \big(\boldsymbol{K}\,\Pi_k^0\,\boldsymbol{u}_h, \Pi_k^0\,\boldsymbol{v}_h\big)_{0,E} + S^E(\boldsymbol{u}_h,\boldsymbol{v}_h) \Big], \tag{26}$$

where the stabilization term $S^E(\cdot,\cdot)$ is any symmetric and positive definite bilinear form that satisfies, $\forall \boldsymbol{v}_h \in \boldsymbol{V}_{h,k}$

$$\alpha_* a_{|E}(\boldsymbol{v}_h,\boldsymbol{v}_h) \le S^E(\boldsymbol{v}_h,\boldsymbol{v}_h) \le \alpha^* a_{|E}(\boldsymbol{v}_h,\boldsymbol{v}_h)$$

for some constants $\alpha_*, \alpha^* > 0$ that are depending on $\boldsymbol{K}$ but independent of $h$. As in [6, 7], we will choose

$$S^E(\boldsymbol{u}_h,\boldsymbol{v}_h) = \overline{\boldsymbol{K}}|E| \sum_{r=1}^{N_E^{\mathrm{dof}}} \mathrm{dof}_r\big((I-\Pi_k^0)\,\boldsymbol{u}_h\big)\mathrm{dof}_r\big((I-\Pi_k^0)\,\boldsymbol{v}_h\big),$$

where $\overline{\boldsymbol{K}}$ is the largest singular value of $\boldsymbol{K}$ on $E$.

Finally, the mixed VEM approximation of (8) is given by:

Find $\boldsymbol{u}_h = \boldsymbol{u}_{0,h} + \boldsymbol{u}_{N,h}$, with $\boldsymbol{u}_{0,h} \in \boldsymbol{V}_{h,k}$, and $p_h \in Q_{h,k}$ such that $\forall \boldsymbol{v}_h \in \boldsymbol{V}_{h,k}$ and $\forall q_h \in Q_{h,k}$:

$$\begin{cases} a_h(\boldsymbol{u}_h,\boldsymbol{v}_h) - (p_h, \nabla\cdot\boldsymbol{v}_h)_{0,\Omega} - \big(\boldsymbol{\beta}p_h, \Pi_k^0\,\boldsymbol{v}_h\big)_{0,\Omega} = -\langle g_D, \boldsymbol{v}_h\cdot\boldsymbol{n}_{\Gamma_D}\rangle_{\pm\frac{1}{2},\Gamma_D} \\ (\nabla\cdot\boldsymbol{u}_h, q_h)_{0,\Omega} + (\gamma p_h, q_h)_{0,\Omega} = (f,q_h)_{0,\Omega} \end{cases} \tag{27}$$

where $\boldsymbol{u}_{N,h} \in \{\boldsymbol{v} \in H(\mathrm{div};\Omega) : \boldsymbol{v}_{|E} \in \boldsymbol{V}_{h,k}(E)\ \forall E \in \mathcal{T}_h\}$ is a proper function that satisfies $\mathrm{dof}_i^e(\boldsymbol{u}_{N,h}) = \mathrm{dof}_i^e(\boldsymbol{u}_N)$, $\forall i = 1,\dots,k+1$ and for each edge $e \in \mathcal{E}_h$, belonging to the Neumann boundary $\Gamma_N$.

The problem (27) has unique solution $(\boldsymbol{u}_h, p_h) \in \boldsymbol{V}_{h,k} \times Q_{h,k}$ and, for $h$ sufficiently small, the following a priori error estimates hold true

$$\|p - p_h\|_0 = O(h^{k+1}), \quad \|\boldsymbol{u} - \boldsymbol{u}_h\|_0 = O(h^{k+1}). \tag{28}$$

Furthermore, the following superconvergence result holds true.

**Theorem 3.1** (Superconvergence result)**.** *Let $p_h$ the solution to (27) and let $p_I \in Q_{h,k}$ be the interpolant of $p$. Then, for $h$ sufficiently small,*

$$\|p_I - p_h\|_0 = O(h^{k+2}). \tag{29}$$

## 4 | POLYNOMIAL BASIS

In this section, we show different procedures for building some polynomial bases for both $\mathbb{P}_k(E)$ and $\big[\mathbb{P}_k(E)\big]^2$. In the following, the left superscript will denote the underlying polynomial basis used for the space $\mathbb{P}_k(E)$. In particular, we will use the symbol $\mathbb{m}$ to indicate the scaled monomial basis (10) and the symbol $\mathbb{q}$ to refer to the orthogonal basis $\{\mathbb{q}_\alpha^k\}_{\alpha=1}^{n_k}$ for $\mathbb{P}_k(E)$, whose construction will be detailed in the following. Finally, we will use the symbol $\mathbb{p}$ to denote a generic polynomial basis $\{\mathbb{p}_\alpha^k\}_{\alpha=1}^{n_k}$ for $\mathbb{P}_k(E)$.

As mentioned in the previous section, the standard choice for the polynomial basis is the set of scaled monomials, defined in (10) (see [3, 4, 7]). Orthogonal polynomial bases have already been

introduced in the virtual element literature, since it has been proven that modifying the definition of internal moments by choosing an $L^2$ $(E)$-orthonormal basis for $\mathbb{P}_k$ $(E)$ can largely improve the reliability of the virtual element method for higher order approximations and in the presence of badly-shaped polygons ([9, 15]).

An efficient procedure for building an $L^2$ $(E)$-orthonormal polynomial basis for $\mathbb{P}_k$ $(E)$ is presented in [2]. This procedure consists in the application of the modified Gram-Schmidt (MGS) orthogonalization process to the monomial Vandermonde matrix ${}^{\mathbb{m}}\mathbf{V}^{E,k} \in \mathbb{R}^{N^{E,Q}\times n_k}$ associated with a quadrature formula $\left\{\left(\boldsymbol{x}_i^{E,Q}, \omega_i^{E,Q}\right)\right\}_{i=1}^{N^{E,Q}}$ on a given element $E$ with $N^{E,Q}$ nodes. As suggested in [2, 13], the process must be applied twice in order to make the orthonomalization error $\|\mathbf{I} - {}^{\mathbb{q}}\mathbf{H}^{E,k}\|$ independent of the condition number of ${}^{\mathbb{m}}\mathbf{V}^{E,k}$ matrix, where ${}^{\mathbb{q}}\mathbf{H}^{E,k} \in \mathbb{R}^{n_k\times n_k}$ is the mass matrix of the resulting $L^2$ $(E)$-orthonormal polynomial basis $\{\mathbb{q}_\alpha^k\}_{\alpha=1}^{n_k}$ for $\mathbb{P}_k$ $(E)$.

In particular, the whole procedure defines a matrix $\mathbf{L}^{E,k} \in \mathbb{R}^{n_k\times n_k}$ on each element $E$ such that

$$ {}^{\mathbb{q}}\mathbf{V}^{E,k} = {}^{\mathbb{m}}\mathbf{V}^{E,k}\left(\mathbf{L}^{E,k}\right)^T, \tag{30} $$

where ${}^{\mathbb{q}}\mathbf{V}^{E,k}$ is the Vandermonde matrix associated with the new polynomial basis. More precisely, we first apply the MGS process to the ${}^{\mathbb{m}}\mathbf{V}^{E,k}$ matrix, that is, we define an upper triangular matrix $\mathbf{R}_1^{E,k} \in \mathbb{R}^{n_k\times n_k}$ and an orthonormal matrix $\mathbf{Q}_1^{E,k} \in \mathbb{R}^{N^{E,Q}\times n_k}$ such that

$$ {}^{\mathbb{m}}\mathbf{V}^{E,k} = \mathbf{Q}_1^{E,k}\mathbf{R}_1^{E,k}. $$

Then we apply MGS to the $\mathbf{Q}_1^{E,k}$ matrix properly rescaled by quadrature weights in order to obtain an $L^2$ $(E)$-orthonormal basis:

$$ \left(\mathbf{Z}^{E,Q}\right)^{1/2}\mathbf{Q}_1^{E,k} = \mathbf{Q}_2^{E,k}\mathbf{R}_2^{E,k}, $$

where $\mathbf{Z}^{E,Q} \in \mathbb{R}^{N^{E,Q}\times N^{E,Q}}$ is the diagonal matrix whose diagonal entries are the quadrature weights. Being $\mathbf{L}^{E,k} = \left(\mathbf{R}_2^{E,k}\mathbf{R}_1^{E,k}\right)^{-T}$, we note that

$$ \begin{aligned} \mathbf{Q}_2^{E,k} &= \left(\mathbf{Z}^{E,Q}\right)^{1/2}\mathbf{Q}_1^{E,k}\left(\mathbf{R}_2^{E,k}\right)^{-1} \\ &= \left(\mathbf{Z}^{E,Q}\right)^{1/2}\ {}^{\mathbb{m}}\mathbf{V}^{E,k}\left(\mathbf{L}^{E,k}\right)^T \\ &= \left(\mathbf{Z}^{E,Q}\right)^{1/2}\ {}^{\mathbb{q}}\mathbf{V}^{E,k} \end{aligned} $$

which means that $\mathbf{Q}_2^{E,k}$ is the Vandermonde matrix ${}^{\mathbb{q}}\mathbf{V}^{E,k}$ rescaled by the square root of the quadrature weights, thus,

$$ {}^{\mathbb{q}}\mathbf{H}^{E,k} = \left({}^{\mathbb{q}}\mathbf{V}^{E,k}\right)^T\mathbf{Z}^{E,Q}\ {}^{\mathbb{q}}\mathbf{V}^{E,k} = \left(\mathbf{Q}_2^{E,k}\right)^T\mathbf{Q}_2^{E,k} = \mathbf{I}. \tag{31} $$

*Remark* 1. Note that ${}^{\mathbb{q}}\mathbf{V}^{E,k} = {}^{\mathbb{m}}\mathbf{V}^{E,k}\left(\mathbf{L}^{E,k}\right)^T$ means that the columns of $\left(\mathbf{L}^{E,k}\right)^T$ are the coefficients that provide each orthonormal polynomial as a linear combination of the scaled monomials.

### 4.1 | Polynomial bases for $\left[\mathbb{P}_k\ (E)\right]^2$

For simplicity, in the following, we will drop the superscript $E$ when no ambiguity occurs. Now, we introduce some auxiliary matrices that we will use in the following. Let ${}^{\mathbb{p}}\boldsymbol{D}_x^{k+1}$, ${}^{\mathbb{p}}\boldsymbol{D}_y^{k+1} \in \mathbb{R}^{n_{k+1}\times n_k}$ be the matrices which collect the coefficients of the partial derivatives of $\{\mathbb{p}_\alpha^{k+1}\}_{\alpha=1}^{n_{k+1}}$, that is,

$$ \frac{\partial \mathbb{p}_\alpha^{k+1}}{\partial x} = \sum_{\beta=1}^{n_k}\left({}^{\mathbb{p}}\boldsymbol{D}_x^{k+1}\right)_{\alpha\beta}\mathbb{p}_\beta^k, \quad \frac{\partial \mathbb{p}_\alpha^{k+1}}{\partial y} = \sum_{\beta=1}^{n_k}\left({}^{\mathbb{p}}\boldsymbol{D}_y^{k+1}\right)_{\alpha\beta}\mathbb{p}_\beta^k, \quad \forall\alpha = 1, \dots, n_{k+1}. \tag{32} $$

Note that, if the MGS basis is used, matrices ${}^{\mathbb{q}}\boldsymbol{\mathcal{D}}_*^{k+1}$ can be derived from the (easily computable) monomial ones ${}^{\mathbb{m}}\boldsymbol{\mathcal{D}}_*^{k+1}$ as:

$$ {}^{\mathbb{q}}\boldsymbol{\mathcal{D}}_*^{k+1} = \mathbf{L}^{k+1}\, {}^{\mathbb{m}}\boldsymbol{\mathcal{D}}_*^{k+1}(\mathbf{L}^k)^{-1} \quad \forall * \in \{x, y\}, $$

since, $\forall \alpha = 1, \dots, n_{k+1}$,

$$ \begin{aligned} \frac{\partial \mathbb{q}_\alpha^{k+1}}{\partial *} &= \sum_{\beta=1}^{n_{k+1}} \mathbf{L}_{\alpha\beta}^{k+1} \frac{\partial \mathbb{m}_\beta^{k+1}}{\partial *} = \sum_{\beta=1}^{n_{k+1}} \sum_{\gamma=1}^{n_k} \mathbf{L}_{\alpha\beta}^{k+1} \left( {}^{\mathbb{m}}\boldsymbol{\mathcal{D}}_*^{k+1} \right)_{\beta\gamma} \mathbb{m}_\gamma^k \\ &= \sum_{\beta=1}^{n_{k+1}} \sum_{\gamma=1}^{n_k} \sum_{s=1}^{n_k} \mathbf{L}_{\alpha\beta}^{k+1} \left( {}^{\mathbb{m}}\boldsymbol{\mathcal{D}}_*^{k+1} \right)_{\beta\gamma} (\mathbf{L}^k)_{\gamma s}^{-1}\ \mathbb{q}_s^k. \end{aligned} $$

*Remark* 2. As highlighted in [2], the modified Gram-Schmidt algorithm allows us to obtain a hierarchical sequence of bases $\{\{\mathbb{q}_\alpha^k\}_{\alpha=1}^{n_k}\}_{k\geq 0}$, that is,

$$ \{\mathbb{q}_\alpha^k\}_{\alpha=1}^{n_k} \subset \{\mathbb{q}_\alpha^{k+1}\}_{\alpha=1}^{n_{k+1}}. \tag{33} $$

As a consequence, we only need to compute $\mathbf{L}^{k+1}$ and then we can set

$$ \mathbf{L}^k = \mathbf{L}^{k+1}(1 : n_k, 1 : n_k), \tag{34} $$

being $\mathbf{L}^{k+1}(1 : n_k, 1 : n_k)$ the matrix obtained from the first $n_k$ rows and columns of $\mathbf{L}^{k+1}$. Indeed, we define the Vandermonde matrices of both $k$ and $k+1$ orders with respect to the same quadrature formula.

Starting from a generic polynomial basis for $\mathbb{P}_k(E)$, as shown in [7], an easily computable basis $\{\boldsymbol{p}_I^k\}_{I=1}^{2n_k}$ for $\left[\mathbb{P}_k(E)\right]^2$ can be built as

$$ \boldsymbol{p}_I^k = \begin{cases} \begin{bmatrix} \mathbb{p}_I^k \\ 0 \end{bmatrix} & I = 1, \dots, n_k \\ \\ \begin{bmatrix} 0 \\ \mathbb{p}_{I-n_k}^k \end{bmatrix} & I = n_k + 1, \dots, 2n_k \end{cases} \tag{35} $$

The Vandermonde matrix ${}^{\mathbb{p}}\mathbf{V}^k \in \mathbb{R}^{2N^{\mathcal{Q}} \times 2n_k}$ associated with the $\{\boldsymbol{p}_I^k\}_{I=1}^{2n_k}$ polynomial basis functions can be written as

$$ {}^{\mathbb{p}}\mathbf{V}^k = \begin{bmatrix} {}^{\mathbb{p}}\mathbf{V}^k & \mathbf{O} \in \mathbb{R}^{N^{\mathcal{Q}} \times n_k} \\ \mathbf{O} \in \mathbb{R}^{N^{\mathcal{Q}} \times n_k} & {}^{\mathbb{p}}\mathbf{V}^k \end{bmatrix}, \tag{36} $$

where

- the top left ${}^{\mathbb{p}}\mathbf{V}^k$ matrix contains the evaluations of the $x$-components of $\boldsymbol{p}_I^k$ for all $I = 1, \dots, n_k$;
- the top right $\mathbf{O}$ matrix contains the evaluations of the $y$-components of $\boldsymbol{p}_I^k$ for all $I = 1, \dots, n_k$, that are all zeros;
- the bottom left $\mathbf{O}$ matrix contains the evaluations of the $x$-components of $\boldsymbol{p}_I^k$ for all $I = n_k + 1, \dots, 2n_k$, that are all zeros;
- the bottom right ${}^{\mathbb{p}}\mathbf{V}^k$ matrix contains the evaluations of the $y$-components of $\boldsymbol{p}_I^k$ for all $I = n_k + 1, \dots, 2n_k$.

Note that if $\{\mathbb{p}_\alpha^k\}_{\alpha=1}^{n_k}$ is an $L^2(E)$-orthonormal polynomial basis for $\mathbb{P}_k(E)$, then $\{\boldsymbol{p}_I^k\}_{I=1}^{2n_k}$ is an $L^2(E)$-orthonormal polynomial basis for $\left[\mathbb{P}_k(E)\right]^2$.

Now, functions belonging to $\mathcal{G}_k^{\nabla}(E)$, as defined in (11), can be written as

$$\boldsymbol{g}_{\alpha}^{\nabla,k} = \nabla \mathbb{p}_{\alpha+1}^{k+1} = \sum_{I=1}^{2n_k} \mathbf{T}_{\alpha I}^{\nabla,k} \boldsymbol{p}_I^k \quad \forall \alpha = 1, \dots, n_k^{\nabla}, \tag{37}$$

where $\mathbf{T}^{\nabla,k} \in \mathbb{R}^{n_k^{\nabla} \times 2n_k}$ is the coefficient matrix of gradients of the polynomial functions $\{\mathbb{p}_{\alpha}^{k+1}\}_{\alpha=2}^{n_{k+1}}$ with respect to the polynomial basis $\{\boldsymbol{p}_I^k\}_{I=1}^{2n_k}$ of $\left[\mathbb{P}_k(E)\right]^2$ and the corresponding Vandermonde matrix is

$${}^{g}\mathbf{V}^{\nabla,k} = {}^{p}\mathbf{V}^{k} \left(\mathbf{T}^{\nabla,k}\right)^T. \tag{38}$$

Based on definitions (32), the $\mathbf{T}^{\nabla,k}$ matrix reads

$$\mathbf{T}^{\nabla,k} = \left[ {}^{\mathbb{p}}\boldsymbol{\mathcal{D}}_x^{k+1}(2:n_{k+1},:) \qquad {}^{\mathbb{p}}\boldsymbol{\mathcal{D}}_y^{k+1}(2:n_{k+1},:) \right],$$

with ${}^{\mathbb{p}}\boldsymbol{\mathcal{D}}_*^{k+1}(2:n_{k+1},:)$ the sub-matrix of ${}^{\mathbb{p}}\boldsymbol{\mathcal{D}}_*^{k+1}$ obtained extracting rows from 2 to $n_{k+1}$ and all columns.

Now, we can complete a basis $\mathcal{G}_k(E)$ for $\left[\mathbb{P}_k(E)\right]^2$ by adding the set of functions $\mathcal{G}_k^{\perp}(E) = \{\boldsymbol{g}_{\alpha}^{\perp,k}\}_{\alpha=1}^{n_k^{\perp}}$ defined in such a way (12) is satisfied. As suggested in [7], $\boldsymbol{g}_{\alpha}^{\perp,k}$ function can be defined as

$$\boldsymbol{g}_{\alpha}^{\perp,k} = \sum_{I=1}^{2n_k} \mathbf{T}_{\alpha I}^{\perp,k} \boldsymbol{p}_I^k \quad \forall \alpha = 1, \dots, n_k^{\perp}, \tag{39}$$

where $\mathbf{T}^{\perp,k} \in \mathbb{R}^{n_k^{\perp} \times 2n_k}$ is the matrix whose rows define an *euclidean* orthonormal basis for the nullspace of $\mathbf{T}^{\nabla,k}$ matrix. Thus, by considering the Singular Value Decomposition of $\mathbf{T}^{\nabla,k} = \mathbf{U}\,\boldsymbol{\Sigma}(\mathbf{V})^T$, we can define $\mathbf{T}^{\perp,k}$ as

$$\mathbf{T}^{\perp,k} = \left[\mathbf{V}(:, n_k^{\nabla}+1 : 2n_k)\right]^T, \tag{40}$$

where $\mathbf{V}(:, n_k^{\nabla}+1 : 2n_k)$ is the submatrix of $\mathbf{V}$ made up of all its rows and of the columns running from the $(n_k^{\nabla}+1)$-th to the $2n_k$-th. As a consequence,

$$\mathbf{T}^{\nabla,k} \left(\mathbf{T}^{\perp,k}\right)^T = \mathbf{O}. \tag{41}$$

The Vandermonde matrix ${}^{g}\mathbf{V}^{k} \in \mathbb{R}^{2N^{Q} \times 2n_k}$ associated with the basis functions $\mathcal{G}_k(E) = \{\boldsymbol{g}_I^k\}_{I=1}^{2n_k} = \{\boldsymbol{g}_{\alpha}^{\nabla,k}\}_{\alpha=1}^{n_k^{\nabla}} \cup \{\boldsymbol{g}_{\beta}^{\perp,k}\}_{\beta=1}^{n_k^{\perp}}$ reads

$${}^{g}\mathbf{V}^{k} = \left[ {}^{g}\mathbf{V}^{\nabla,k} \qquad {}^{g}\mathbf{V}^{\perp,k} \right] = {}^{p}\mathbf{V}^{k} \left[ \left(\mathbf{T}^{\nabla,k}\right)^T \qquad \left(\mathbf{T}^{\perp,k}\right)^T \right]. \tag{42}$$

We define $\mathbf{G}^{k} \in \mathbb{R}^{2n_k \times 2n_k}$ as the mass matrix related to the $\mathcal{G}_k(E)$ basis, whose entries are given by

$$\mathbf{G}_{IJ}^{k} = \int_E \boldsymbol{g}_I^k \cdot \boldsymbol{g}_J^k, \quad \forall I, J = 1, \dots, 2n_k. \tag{43}$$

In matrix form, $\mathbf{G}^{k}$ reads

$$\begin{aligned} \mathbf{G}^{k} &= \left({}^{g}\mathbf{V}^{k}\right)^T \begin{bmatrix} \mathbf{Z}^{Q} & \mathbf{O} \\ \mathbf{O} & \mathbf{Z}^{Q} \end{bmatrix} {}^{g}\mathbf{V}^{k} \\ &= \begin{bmatrix} \mathbf{T}^{\nabla,k}\, {}^{p}\mathbf{H}^{k} \left(\mathbf{T}^{\nabla,k}\right)^T & \mathbf{T}^{\nabla,k}\, {}^{p}\mathbf{H}^{k} \left(\mathbf{T}^{\perp,k}\right)^T \\ \mathbf{T}^{\perp,k}\, {}^{p}\mathbf{H}^{k} \left(\mathbf{T}^{\nabla,k}\right)^T & \mathbf{T}^{\perp,k}\, {}^{p}\mathbf{H}^{k} \left(\mathbf{T}^{\perp,k}\right)^T \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{G}^{\nabla,\nabla} & \mathbf{G}^{\nabla,\perp} \\ \mathbf{G}^{\perp,\nabla} & \mathbf{G}^{\perp,\perp} \end{bmatrix}, \end{aligned} \tag{44}$$

where

$$ {}^{p}\mathbf{H}^{k} = \left({}^{p}\mathbf{V}^{k}\right)^{T} \begin{bmatrix} \mathbf{Z}^{Q} & \mathbf{O} \\ \mathbf{O} & \mathbf{Z}^{Q} \end{bmatrix} {}^{p}\mathbf{V}^{k} $$

is the mass matrix related to $\{\boldsymbol{p}_I\}_{I=1}^{2n_k}$ basis.

Now, we make the following observations.

**1** If we choose the standard set of scaled monomials $\mathcal{M}_k(E)$ as the basis for $\mathbb{P}_k(E)$ and we set

$$ {}^{1}\mathbf{T}^{\nabla,k} = \left[ {}^{\mathbb{m}}\boldsymbol{D}_x^{k+1}(2:n_{k+1},:) \quad {}^{\mathbb{m}}\boldsymbol{D}_y^{k+1}(2:n_{k+1},:) \right] = {}^{1}\mathbf{U}\,{}^{1}\boldsymbol{\Sigma}\left({}^{1}\mathbf{V}\right)^{T}, \quad {}^{1}\mathbf{T}^{\perp,k} = \left[{}^{1}\mathbf{V}(:,n_k^{\nabla}+1:2n_k)\right]^{T}, $$

it is known that the corresponding ${}^{1}\mathbf{G}^{k}$ matrix, that is,

$$ {}^{1}\mathbf{G}^{k} = \begin{bmatrix} {}^{1}\mathbf{T}^{\nabla,k}\,{}^{m}\mathbf{H}^{k}\left({}^{1}\mathbf{T}^{\nabla,k}\right)^{T} & {}^{1}\mathbf{T}^{\nabla,k}\,{}^{m}\mathbf{H}^{k}\left({}^{1}\mathbf{T}^{\perp,k}\right)^{T} \\ {}^{1}\mathbf{T}^{\perp,k}\,{}^{m}\mathbf{H}^{k}\left({}^{1}\mathbf{T}^{\nabla,k}\right)^{T} & {}^{1}\mathbf{T}^{\perp,k}\,{}^{m}\mathbf{H}^{k}\left({}^{1}\mathbf{T}^{\perp,k}\right)^{T} \end{bmatrix} \tag{45} $$

will be ill-conditioned for high polynomial degrees. Anyway, we want to highlight that, in this so-called *monomial approach*, the rows of ${}^{1}\mathbf{T}^{\perp,k}$ are orthonormal to each other and are orthogonal to the rows of ${}^{1}\mathbf{T}^{\nabla,k}$ with respect to the euclidean scalar product, by construction.

**2** On the other hand, if we choose the MGS basis $\{\mathbb{q}_\alpha^k\}_{\alpha=1}^{n_k}$, that is, the $L^2(E)$-orthonormal polynomial basis for $\mathbb{P}_k(E)$ introduced in [2], the corresponding mass matrix ${}^{q}\mathbf{H}^{k}$ will be the identity matrix $\mathbf{I}$. Thus, by setting

$$ {}^{2}\mathbf{T}^{\nabla,k} = \left[ {}^{\mathbb{q}}\boldsymbol{D}_x^{k+1}(2:n_{k+1},:) \quad {}^{\mathbb{q}}\boldsymbol{D}_y^{k+1}(2:n_{k+1},:) \right] = {}^{2}\mathbf{U}\,{}^{2}\boldsymbol{\Sigma}\left({}^{2}\mathbf{V}\right)^{T}, \quad {}^{2}\mathbf{T}^{\perp,k} = \left[{}^{2}\mathbf{V}(:,n_k^{\nabla}+1:2n_k)\right]^{T}, \tag{46} $$

the related ${}^{2}\mathbf{G}^{k}$ matrix, in infinite precision, is given by

$$ {}^{2}\mathbf{G}^{k} = \begin{bmatrix} {}^{2}\mathbf{T}^{\nabla,k}\,\mathbf{I}\left({}^{2}\mathbf{T}^{\nabla,k}\right)^{T} & {}^{2}\mathbf{T}^{\nabla,k}\,\mathbf{I}\left({}^{2}\mathbf{T}^{\perp,k}\right)^{T} \\ {}^{2}\mathbf{T}^{\perp,k}\,\mathbf{I}\left({}^{2}\mathbf{T}^{\nabla,k}\right)^{T} & {}^{2}\mathbf{T}^{\perp,k}\,\mathbf{I}\left({}^{2}\mathbf{T}^{\perp,k}\right)^{T} \end{bmatrix} = \begin{bmatrix} {}^{2}\mathbf{T}^{\nabla,k}\left({}^{2}\mathbf{T}^{\nabla,k}\right)^{T} & \mathbf{O} \\ \mathbf{O} & \mathbf{I} \end{bmatrix}, \tag{47} $$

where the last equation is a consequence of property (41). In conclusion, if an $L^2(E)$-orthonormal polynomial basis for $\mathbb{P}_k(E)$ is used, $\mathbf{2}_{\mathcal{G}}^{k}(E)$ will be partially $L^2(E)$-orthonormalized, since the ${}^{2}\boldsymbol{g}_\alpha^{\perp,k}$ functions are a set of $L^2(E)$-orthonormal functions and are $L^2(E)$-orthogonal to ${}^{2}\boldsymbol{g}_\alpha^{\nabla,k}$ functions, but the ${}^{2}\boldsymbol{g}_\alpha^{\nabla,k}$ are not naturally $L^2(E)$-orthogonal to each other. We denote this as *partial-orthonormal approach*.

#### 4.1.1 | A full $L^2(E)$-orthonormal approach

In this section, we will show a procedure that allows to full $L^2(E)$-orthonormalize the $\mathcal{G}_k(E)$ basis. For this purpose, we first note from (47) that in order to make $\mathcal{G}_k^{\nabla}(E)$ an $L^2(E)$-orthonormal set of functions, it is sufficient to orthonormalize the rows of ${}^{2}\mathbf{T}^{\nabla,k}$ matrix defined in (46) with respect to the euclidean scalar product. In order to do this, we apply only once the modified Gram-Schmidt algorithm to $\left({}^{2}\mathbf{T}^{\nabla,k}\right)^{T}$. More precisely, we factorize $\left({}^{2}\mathbf{T}^{\nabla,k}\right)^{T}$ as

$$ \left({}^{2}\mathbf{T}^{\nabla,k}\right)^{T} = \mathbf{Q}^{\nabla,k}\mathbf{R}^{\nabla,k}, $$

and, then we set

$$ {}^{3}\mathbf{T}^{\nabla,k} = (\mathbf{Q}^{\nabla,k})^{T} = \mathbf{L}^{\nabla,k}\,{}^{2}\mathbf{T}^{\nabla,k}, \quad {}^{3}\mathbf{T}^{\perp,k} = {}^{3}\mathbf{V}(:,n_k^{\nabla}+1:2n_k)^{T}, \tag{48} $$

where ${}^{3}\mathbf{T}^{\nabla,k} = {}^{3}\mathbf{U}{}^{3}\boldsymbol{\Sigma}{}^{3}\mathbf{V}^{T}$ and $\mathbf{L}^{\nabla,k} = \left(\mathbf{R}^{\nabla,k}\right)^{-T} \in \mathbb{R}^{n_k^{\nabla}\times n_k^{\nabla}}$.

By proceeding in this way, the ${}^{3}\mathbf{G}^{k}$ matrix will become the identity matrix. We will call this procedure as *full-orthonormal approach*.

*Remark* 3. It is worth mentioning that in order to orthonormalize the rows of ${}^{2}\mathbf{T}^{\nabla,k}$ we could resort to its computed Singular Value Decomposition (46) and set

$$
\begin{aligned}
{}^{3}\mathbf{T}^{\nabla,k} &= {}^{2}\mathbf{V}(:,1:n_k^{\nabla})^{T} = \left({}^{2}\mathbf{\Sigma}(:,1:n_k^{\nabla})\right)^{-1}{}^{2}\mathbf{U}^{T}\ {}^{2}\mathbf{T}^{\nabla,k},\\
{}^{3}\mathbf{T}^{\perp,k} &= {}^{2}\mathbf{V}(:,n_k^{\nabla}+1:2n_k)^{T} = {}^{2}\mathbf{T}^{\perp,k}.
\end{aligned}
$$

In this way, we obtain that ${}^{3}\mathbf{G}^{k} = \mathbf{I}$, but the SVD process is not hierarchical and, as a consequence, the set of functions $\{{}^{3}\boldsymbol{g}_{\alpha}^{\nabla,k-1}\}_{\alpha=1}^{n_{k-1}^{\nabla}}$ used for defining the internal $\nabla$ dofs is not $L^2(E)$-orthonormalized for free. As highlighted in [9, 15], this type of change can improve only the condition number of elemental matrices, but does not ensure to improve the global performance of the method.

Instead, since the MGS is a hierarchical procedure (see Remark 2), we obtain that the functions

$$
\left\{{}^{3}\boldsymbol{g}_{\alpha}^{\nabla,k-1}\right\}_{\alpha=1}^{n_{k-1}^{\nabla}} \subset \left\{{}^{3}\boldsymbol{g}_{\alpha}^{\nabla,k}\right\}_{\alpha=1}^{n_{k}^{\nabla}} \tag{49}
$$

used in the definition of internal $\nabla$ dofs are a set of $L^2$ $(E)$-orthonormal functions.

## 5 | SOME IMPLEMENTATION DETAILS

In this section, we will show how to compute the local matrices needed for assembling the local system matrix related to the discrete problem (27) with the aforementioned approaches, following a procedure similar to the one shown in [7].

### 5.1 | The divergence term $(\nabla\cdot\boldsymbol{u}_h, q_h)_{\mathbf{0},E}$

By choosing $\boldsymbol{u}_h = \boldsymbol{\varphi}_i$ and $q_h = \mathbb{p}_{\alpha}^{k}$, we define $\mathbf{W} \in \mathbb{R}^{n_k \times N_E^{\mathrm{dof}}}$ as the matrix whose entries read, $\forall i = 1, \dots, N_E^{\mathrm{dof}}$, $\alpha = 1, \dots, n_k$,

$$
\mathbf{W}_{\alpha i} = \int_E \nabla\cdot\boldsymbol{\varphi}_i\ \mathbb{p}_{\alpha}^{k} \tag{50}
$$

$$
\begin{aligned}
&= -\int_E \boldsymbol{\varphi}_i\cdot\nabla\mathbb{p}_{\alpha}^{k} + \int_{\partial E}\boldsymbol{\varphi}_i\cdot\boldsymbol{n}_{\partial E}\ \mathbb{p}_{\alpha}^{k}\\
&:= (\mathbf{W}_1)_{\alpha i} + (\mathbf{W}_2)_{\alpha i}.
\end{aligned} \tag{51}
$$

Since in the monomial and partial-orthonormal approaches we have

$$
\begin{aligned}
\mathbb{p} = \mathbb{m}, \quad \nabla\mathbb{m}_{\alpha}^{k} &= {}^{1}\boldsymbol{g}_{\alpha-1}^{\nabla,k-1} \quad \forall\alpha = 2, \dots, n_k,\\
\mathbb{p} = \mathbb{q}, \quad \nabla\mathbb{q}_{\alpha}^{k} &= {}^{2}\boldsymbol{g}_{\alpha-1}^{\nabla,k-1} \quad \forall\alpha = 2, \dots, n_k,
\end{aligned} \tag{52}
$$

by exploiting the internal $\nabla$ degrees of freedom (18), we have

$$
\mathbf{W}_1 = \begin{bmatrix} & \mathbf{O}\in\mathbb{R}^{1\times N_E^{\mathrm{dof}}} & \\ \mathbf{O}\in\mathbb{R}^{n_{k-1}^{\nabla}\times N^{E,e}(k+1)} & -|E|\ \mathbf{I}\in\mathbb{R}^{n_{k-1}^{\nabla}\times n_{k-1}^{\nabla}} & \mathbf{O}\in\mathbb{R}^{n_{k-1}^{\nabla}\times n_{k}^{\perp}} \end{bmatrix}.
$$

Concerning the full-orthonormal approach, we observe that

$$\begin{aligned}\mathbb{p} = \mathbb{q}, \quad \nabla \mathbb{q}^k_{\alpha+1} &= \sum_{I=1}^{2n_{k-1}} {}^{2}\mathbf{T}^{\nabla,k}_{\alpha I}\, \boldsymbol{q}^{k-1}_I = \sum_{I=1}^{2n_{k-1}} \sum_{\beta=1}^{n^{\nabla}_{k-1}} \left(\mathbf{L}^{\nabla,k-1}\right)^{-1}_{\alpha\beta} {}^{3}\mathbf{T}^{\nabla,k}_{\beta I}\, \boldsymbol{q}^{k-1}_I \\ &= \sum_{\beta=1}^{n^{\nabla}_{k-1}} \left(\mathbf{L}^{\nabla,k-1}\right)^{-1}_{\alpha\beta} {}^{3}\boldsymbol{g}^{\nabla,k-1}_{\beta}, \quad \forall \alpha = 1, \dots, n^{\nabla}_{k-1}.\end{aligned} \tag{53}$$

Thus, thanks to the hierarchical property shown in Remark 3, the $\mathbf{W}_1$ matrix becomes

$$\mathbf{W}_1 = \begin{bmatrix} \mathbf{O} \in \mathbb{R}^{1\times N^{\text{dof}}_E} \\ \mathbf{O} \in \mathbb{R}^{n^{\nabla}_{k-1}\times N^{E,e}(k+1)} \quad -|E|\left(\mathbf{L}^{\nabla,k}\right)^{-1}(1:n^{\nabla}_{k-1}, 1:n^{\nabla}_{k-1}) \quad \mathbf{O} \in \mathbb{R}^{n^{\nabla}_{k-1}\times n^{\perp}_k} \end{bmatrix}.$$

Concerning the second term in the right-hand side of Equation (51), we observe that the integrand function is a known polynomial of degree $2k$ on each edge of $E$, which can be computed exactly (up to machine precision) by exploiting the edge dofs (17). Then, the second term $\mathbf{W}_2$ reads

$$\mathbf{W}_2 = \left[ \left({}^{\mathbb{p}}\mathbf{V}^{\partial,k}\right)^T \mathbf{Z}^{\partial,Q} \quad \mathbf{O} \in \mathbb{R}^{n_k\times(n^{\nabla}_{k-1}+n^{\perp}_k)} \right],$$

where ${}^{p}\mathbf{V}^{\partial,k} \in \mathbb{R}^{N^{E,e}(k+1)\times n_k}$ is the Vandermonde matrix related to the polynomials $\{\mathbb{p}^k_\alpha\}_{\alpha=1}^{n_k}$ and the boundary quadrature formula $\left\{ \left\{ \left( \boldsymbol{x}^{e,Q}_i, \omega^{e,Q}_i \right) \right\}_{i=1}^{k+1} \right\}_{e\in\mathcal{E}_{h,E}}$ defined on $\partial E$, whose quadrature nodes coincide with the nodes used in the edge dofs (17) and the matrix $\mathbf{Z}^{\partial,Q} \in \mathbb{R}^{N^{E,e}(k+1)\times N^{E,e}(k+1)}$ is the diagonal matrix whose non-zero entries coincide with the related boundary quadrature weights properly arranged.

### 5.2 | Computation of $L^2(E)$-projection of basis functions of $V_{h,k}(E)$

Now, we compute the projection $\Pi^0_k \boldsymbol{\varphi}_i$ of the Lagrangian basis functions of $\boldsymbol{V}_{h,k}(E)$ in terms of the $\mathcal{G}_k(E)$ basis, that is, we determine the matrix $\mathbf{\Pi}^{\mathbf{0}}_{\mathbf{k}} \in \mathbb{R}^{2n_k\times N^{\text{dof}}_E}$ such that

$$\Pi^k_0 \boldsymbol{\varphi}_i = \sum_{I=1}^{2n_k} \left(\mathbf{\Pi}^{\mathbf{0}}_{\mathbf{k}}\right)_{Ii} \boldsymbol{g}^k_I. \tag{54}$$

By replacing (54) in the definition (23) of $\Pi^k_0$, we obtain, $\forall i = 1, \dots, N^{\text{dof}}_E$, $\forall J = 1, \dots, 2n_k$,

$$\sum_{I=1}^{2n_k} \left(\mathbf{\Pi}^{\mathbf{0}}_{\mathbf{k}}\right)_{Ii} \left(\boldsymbol{g}^k_I, \boldsymbol{g}^k_J\right)_{0,E} = \left(\boldsymbol{\varphi}_i, \boldsymbol{g}^k_J\right)_{0,E}.$$

or, analogously, in matrix form

$$\mathbf{G}^k \mathbf{\Pi}^{\mathbf{0}}_{\mathbf{k}} = \mathbf{B}, \tag{55}$$

where $\mathbf{B} \in \mathbb{R}^{2n_k\times N^{\text{dof}}_E}$ is the matrix whose entries are defined as

$$\mathbf{B}_{Ji} = \left(\boldsymbol{\varphi}_i, \boldsymbol{g}^k_J\right)_{0,E}, \quad \forall i = 1, \dots, N^{\text{dof}}_E, \ \forall J = 1, \dots, 2n_k.$$

Now, as in [7], we split $\mathbf{B}$ as

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}^{\nabla} \\ \mathbf{B}^{\perp} \end{bmatrix}.$$

The term $\mathbf{B}^{\perp} \in \mathbb{R}^{n_k^{\perp} \times N_E^{\text{dof}}}$, whose entries are

$$\mathbf{B}^{\perp}_{\alpha i} = \left(\boldsymbol{\varphi}_i, \boldsymbol{g}_{\alpha}^{\perp,k}\right)_{0,E}, \quad \forall \alpha = 1, \dots, n_k^{\perp}, \ \forall i = 1, \dots, N_E^{\text{dof}},$$

can be readily computed as

$$\mathbf{B}^{\perp} = \left[\mathbf{O} \in \mathbb{R}^{n_k^{\perp} \times (N^{E,e}(k+1)+n_{k-1}^{\nabla})} \qquad |E|\, \mathbf{I} \in R^{n_k^{\perp} \times n_k^{\perp}}\right],$$

by exploiting the internal $\perp$ degrees of freedom (19).

Regarding the first term $\mathbf{B}^{\nabla} \in \mathbb{R}^{n_k^{\nabla} \times N_E^{\text{dof}}}$, we note that, $\forall \alpha = 1, \dots, n_k^{\nabla}$, $\forall i = 1, \dots, N_E^{\text{dof}}$

- in the monomial and in the partial-orthonormal approaches, recalling Equation (52),

$$\begin{aligned} \mathbf{B}^{\nabla}_{\alpha i} &= \int_E \boldsymbol{\varphi}_i \cdot \boldsymbol{g}_{\alpha}^{\nabla,k} = \int_E \boldsymbol{\varphi}_i \cdot \nabla \mathbb{p}_{\alpha+1}^{k+1} \\ &= -\int_E \nabla \cdot \boldsymbol{\varphi}_i \, \mathbb{p}_{\alpha+1}^{k+1} + \int_{\partial E} \boldsymbol{\varphi}_i \cdot \boldsymbol{n}_{\partial E} \, \mathbb{p}_{\alpha+1}^{k+1} \\ &:= \left(\mathbf{B}_1^{\nabla}\right)_{\alpha i} + \left(\mathbf{B}_2^{\nabla}\right)_{\alpha i}, \end{aligned} \tag{56}$$

- in the full-orthonormal approach, evoking Equation (53), we have

$$\mathbf{B}^{\nabla}_{\alpha i} = \int_E \boldsymbol{\varphi}_i \cdot \boldsymbol{g}_{\alpha}^{\nabla,k} = \sum_{\beta} \mathbf{L}^{\nabla,k}_{\alpha\beta} \int_E \boldsymbol{\varphi}_i \cdot \nabla \mathbb{p}_{\beta+1}^{k+1} = \left(\mathbf{L}^{\nabla,k} \, \mathbf{B}_1^{\nabla}\right)_{\alpha i} + \left(\mathbf{L}^{\nabla,k} \, \mathbf{B}_2^{\nabla}\right)_{\alpha i}.$$

Since the integrand of the second term of (56) is a known polynomial of degree $2k+1$ on each edge of $E$, it can be integrated exactly by exploiting the edge dofs (17). Thus,

$$\mathbf{B}_2^{\nabla} = \left[\left({}^{\mathbb{p}}\mathbf{V}^{\partial,k+1}(:, 2 : n_{k+1})\right)^T \mathbf{Z}^{\partial,Q} \qquad \mathbf{O} \in \mathbb{R}^{n_k^{\nabla} \times (n_{k-1}^{\nabla} + n_k^{\perp})}\right].$$

In order to compute the term $\mathbf{B}_1^{\nabla}$, we first note that $\nabla \cdot \boldsymbol{\varphi}_i$ is a known polynomial of degree $k$ on $E$. Then, it can be written as

$$\nabla \cdot \boldsymbol{\varphi}_i = \sum_{\alpha=1}^{n_k} \boldsymbol{\Lambda}_{\alpha i} \, \mathbb{p}_{\alpha}^{k}, \quad \forall i = 1, \dots, N_E^{\text{dof}}.$$

whose coefficient matrix $\boldsymbol{\Lambda} \in \mathbb{R}^{n_k \times N_E^{\text{dof}}}$ can be retrieved from

$$\sum_{\alpha=1}^{n_k} \boldsymbol{\Lambda}_{\alpha i} \int_E \mathbb{p}_{\alpha}^{k} \mathbb{p}_{\beta}^{k} = \int_E \nabla \cdot \boldsymbol{\varphi}_i \, \mathbb{p}_{\beta}^{k}, \quad \forall \beta = 1, \dots, n_k, \ \forall i = 1, \dots, N_E^{\text{dof}}. \tag{57}$$

Equation (57) can be written in matrix form as

$${}^{\mathbb{p}}\mathbf{H}^{k} \boldsymbol{\Lambda} = \mathbf{W}, \tag{58}$$

since the term at right hand side of (57) coincides with the definition of the $\mathbf{W}$ matrix given in Section 5.1. We highlight that, in the partial and in the full orthonormal approaches, in infinite precision, it holds

$$\boldsymbol{\Lambda} = \mathbf{W}.$$

Finally, matrix $\mathbf{B}_1^{\nabla}$ is given by

$$\left(\mathbf{B}_1^{\nabla}\right)_{\alpha i} = -\int_E \nabla \cdot \boldsymbol{\varphi}_i \mathbb{p}_{\alpha+1}^{k+1} = -\sum_{\beta=1}^{n_k} \boldsymbol{\Lambda}_{\beta i} \int_E \mathbb{p}_{\beta}^{k} \mathbb{p}_{\alpha+1}^{k+1},$$

or, equivalently,

$$\mathbf{B}_1^{\nabla} = -\,{}^{\mathbb{P}}\mathbf{H}^{k+1}(2:n_{k+1},1:n_k)\mathbf{\Lambda} = -\,{}^{\mathbb{P}}\mathbf{H}^{k+1}(2:n_{k+1},1:n_k)\left({}^{\mathbb{P}}\mathbf{H}^{k}\right)^{-1}\mathbf{W}.$$

### 5.3 | The diffusion term $a_h^E(\boldsymbol{u}_h, \boldsymbol{v}_h)$

The local discrete diffusion term, defined in Equation (26), is the sum of a consistency and a stability term. Thus, as in [7], we can separately define the consistency matrix

$$(\mathbf{K}_C^a)_{ij} = \left(\boldsymbol{K}\Pi_0^k\boldsymbol{\varphi}_i, \Pi_0^k\boldsymbol{\varphi}_j\right)_{0,E}$$

and the stability matrix

$$(\mathbf{K}_S^a)_{ij} = \overline{\boldsymbol{K}}|E|\sum_{r=1}^{N_E^{\mathrm{dof}}}\mathrm{dof}_r\left((I-\Pi_0^k)\,\boldsymbol{\varphi}_i\right)\mathrm{dof}_r\left((I-\Pi_0^k)\,\boldsymbol{\varphi}_j\right).$$

By writing the tensor $\boldsymbol{K} = \begin{bmatrix} K_{xx} & K_{xy} \\ K_{xy} & K_{yy} \end{bmatrix}$, we define the matrix $\mathbf{Z}^{Q,K_*} \in \mathbb{R}^{N^Q\times N^Q}$ as the diagonal matrix whose non zeros entries are

$$\mathbf{Z}_{ii}^{Q,K_*} = K_*(\boldsymbol{x}_i^Q)\,\omega_i^Q, \quad \forall i = 1,\ldots,N^Q\ \forall *\in\{xx,xy,yy\}.$$

Then, the consistency matrix can be computed as

$$\mathbf{K}_C^a = \left(\mathbf{\Pi}_{\mathbf{k}}^{\mathbf{0}}\right)^T\mathbf{G}^{k,\boldsymbol{K}}\,\mathbf{\Pi}_{\mathbf{k}}^{\mathbf{0}},$$

where the matrix $\mathbf{G}^{k,\boldsymbol{K}} \in \mathbb{R}^{2n_k\times 2n_k}$ reads

$$\mathbf{G}^{k,\boldsymbol{K}} = \left({}^{\boldsymbol{g}}\mathbf{V}^k\right)^T\begin{bmatrix}\mathbf{Z}^{Q,K_{xx}} & \mathbf{Z}^{Q,K_{xy}} \\ \mathbf{Z}^{Q,K_{xy}} & \mathbf{Z}^{Q,K_{yy}}\end{bmatrix}{}^{\boldsymbol{g}}\mathbf{V}^k,$$

that is, $\mathbf{G}^{k,\boldsymbol{K}}$ is the mass matrix $\mathbf{G}^k$ related to the basis polynomial basis $\mathcal{G}_k\ (E)$ of degree $k$ weighted by the tensor $\boldsymbol{K}$.

Now, we define the matrix $\mathbf{D} \in \mathbb{R}^{N_E^{\mathrm{dof}}\times 2n_k}$, whose entries are

$$\mathbf{D}_{iI} = \mathrm{dof}_i(\boldsymbol{g}_I^k), \quad \forall i = 1,\ldots,N_E^{\mathrm{dof}},\ \forall I = 1,\ldots,2n_k.$$

Matrix $\mathbf{D}$ can be split as

$$\mathbf{D} = \begin{bmatrix}\mathbf{D}^{\partial} \\ \mathbf{D}^{\nabla} \\ \mathbf{D}^{\perp}\end{bmatrix},$$

being matrices $\mathbf{D}^{\nabla} \in \mathbb{R}^{n_{k-1}^{\nabla}\times 2n_k}$ and $\mathbf{D}^{\perp} \in \mathbb{R}^{n_k^{\perp}\times 2n_k}$ given by

$$\mathbf{D}^{\nabla} = \frac{1}{|E|}\mathbf{G}^k(1:n_{k-1}^{\nabla},:), \quad \mathbf{D}^{\perp} = \frac{1}{|E|}\mathbf{G}^k(n_k^{\nabla}+1:2n_k,:).$$

Matrix $\mathbf{D}^{\partial} \in \mathbb{R}^{N^{E,e}(k+1)\times 2n_k}$, instead, $\forall i = 1,\ldots,k+1$, $\forall e \in \mathcal{E}_{h,E}$, $\forall I = 1,\ldots,2n_k,$, reads

$$\mathbf{D}[\partial]_{iI} = \mathrm{dof}_i^e(\boldsymbol{g}_I^k) = \left(\boldsymbol{g}_I^k\cdot\boldsymbol{n}_e\right)(\boldsymbol{x}_i^{e,Q}),$$

or, equivalently,

$$\mathbf{D}^{\partial} = \begin{bmatrix} \mathbf{N}_x & \mathbf{N}_y \end{bmatrix} {}^{g}\mathbf{V}^{\partial,k}$$
$$= \begin{bmatrix} \mathbf{N}_x & \mathbf{N}_y \end{bmatrix} \begin{bmatrix} {}^{\mathbb{P}}\mathbf{V}^{\partial,k} & \mathbf{O} \in \mathbb{R}^{N^{E,e}(k+1)\times n_k} \\ \mathbf{O} \in \mathbb{R}^{N^{E,e}(k+1)\times n_k} & {}^{\mathbb{P}}\mathbf{V}^{\partial,k} \end{bmatrix} \begin{bmatrix} \left(\mathbf{T}^{\nabla,k}\right)^T & \left(\mathbf{T}^{\perp,k}\right)^T \end{bmatrix},$$

where $\mathbf{N}_* \in \mathbb{R}^{N^{E,e}(k+1)\times N^{E,e}(k+1)}$, for all $*\in \{x, y\}$, is the diagonal matrix whose diagonal entries are the $*$-component of the normal vectors to the edges of $E$, properly arranged.

Finally, the stability matrix can be computed as

$$\mathbf{K}_S^a = \overline{K}\ |E|\ \left(\mathbf{I} - \mathbf{D}\ \mathbf{\Pi}_{\mathbf{k}}^{\mathbf{0}}\right)^T \left(\mathbf{I} - \mathbf{D}\ \mathbf{\Pi}_{\mathbf{k}}^{\mathbf{0}}\right).$$

### 5.4 | The advection term $-\left(\boldsymbol{\beta} p_h, \Pi_{\mathbf{0}}^k \boldsymbol{v}_h\right)_{\mathbf{0},E}$

The matrix $\mathbf{A}^{\boldsymbol{\beta}} \in \mathbb{R}^{N_E^{\text{dof}}\times n_k}$ corresponding to the local advection term

$$-(\boldsymbol{\beta} p_h, \boldsymbol{v}_h)_{0,E},$$

$\forall i = 1, \dots, N_E^{\text{dof}}, \forall \alpha = 1, \dots, n_k$, is defined as

$$\mathbf{A}_{i\alpha}^{\boldsymbol{\beta}} = -\int_E \boldsymbol{\beta} \cdot \Pi_0^k \boldsymbol{\varphi}_i\ \mathbb{p}_\alpha^k = -\sum_{I=1}^{2n_k} \left(\mathbf{\Pi}_{\mathbf{k}}^{\mathbf{0}}\right)_{Ii} \int_E \boldsymbol{\beta} \cdot \boldsymbol{g}_I^k\ \mathbb{p}_\alpha^k.$$

Given $\boldsymbol{\beta} = \left[\beta_x, \beta_y\right]^T$, in matrix form $\mathbf{A}^{\boldsymbol{\beta}}$ reads

$$\mathbf{A}^{\boldsymbol{\beta}} = \left(\mathbf{\Pi}_{\mathbf{k}}^{\mathbf{0}}\right)^T \left({}^{g}\mathbf{V}^k\right)^T \begin{bmatrix} \mathbf{Z}^{Q,\beta_x} & \mathbf{O} \\ \mathbf{O} & \mathbf{Z}^{Q,\beta_y} \end{bmatrix} \begin{bmatrix} {}^{\mathbb{P}}\mathbf{V}^k \\ {}^{\mathbb{P}}\mathbf{V}^k \end{bmatrix},$$

where $\mathbf{Z}^{Q,\beta_*} \in \mathbb{R}^{N^Q\times N^Q}\ \forall *\in \{x, y\}$ is the diagonal matrix whose non zero entries are defined as

$$(\mathbf{Z}^{Q,\beta_*})_{ii} = \beta_*(\boldsymbol{x}_i^Q)\ \omega_i^Q, \quad \forall i = 1, \dots, N^Q.$$

### 5.5 | The reaction term $(\gamma p_h, q_h)_{\mathbf{0},E}$

We define the local reaction matrix ${}^{\mathbb{P}}\mathbf{H}^{k,\gamma} \in \mathbb{R}^{n_k\times n_k}$ as the matrix that collects the terms

$${}^{\mathbb{P}}\mathbf{H}_{\alpha\beta}^{k,\gamma} = \int_E \gamma\ \mathbb{p}_\alpha^k\ \mathbb{p}_\beta^k \quad \forall \alpha, \beta = 1, \dots, n_k.$$

We observe that this matrix is the mass matrix ${}^{\mathbb{P}}\mathbf{H}^k$ related to the polynomial basis $\{\mathbb{p}_\alpha^k\}_{\alpha=1}^{n_k}$ of degree $k$ weighted by the reaction coefficient $\gamma$. Thus, in matrix form, we have

$${}^{\mathbb{P}}\mathbf{H}^{k,\gamma} = \left({}^{\mathbb{P}}\mathbf{V}^k\right)^T \mathbf{Z}^{Q,\gamma}\ {}^{\mathbb{P}}\mathbf{V}^k,$$

where

$$\mathbf{Z}_{ij}^{Q,\gamma} = \gamma(\boldsymbol{x}_i^Q)\ \omega_i^Q\ \delta_{ij}, \quad \forall i, j = 1, \dots, N^Q.$$

In conclusion, the local system matrix $\mathbf{K}^E \in \mathbb{R}^{N_E^{\text{dof}}\times N_E^{\text{dof}}}$ for the discrete problem (27) is given by

$$\mathbf{K}^E = \begin{bmatrix} \mathbf{K}_C^a + \mathbf{K}_S^a & -\mathbf{W}^T + \mathbf{A}^{\boldsymbol{\beta}} \\ \mathbf{W} & {}^{p}\mathbf{H}^{k,\gamma} \end{bmatrix}.$$

## 6 | NUMERICAL EXPERIMENTS

In this section, we propose some numerical experiments to show the performance of the aforementioned approaches in terms of the following error norms:

$$p_{\text{err}} = \left( \sum_{E \in \mathcal{T}_h} \|p - p_h\|_{0,E}^2 \right)^{\frac{1}{2}}, \tag{59}$$

$$\boldsymbol{u}_{\text{err}} = \left( \sum_{E \in \mathcal{T}_h} \|\boldsymbol{u} - \Pi_0^k \boldsymbol{u}_h\|_{0,E}^2 \right)^{\frac{1}{2}}, \tag{60}$$

$$p_{I,\text{err}} = \left( \sum_{E \in \mathcal{T}_h} \|p_I - p_h\|_{0,E}^2 \right)^{\frac{1}{2}}, \tag{61}$$

where the interpolant $p_I \in Q_{h,k}$ of $p$ is computed locally as

$$p_I = \sum_{\alpha=1}^{n_k} \boldsymbol{c}_\alpha \mathbb{p}_\alpha^k.$$

The vector $\boldsymbol{c} \in \mathbb{R}^{n_k}$ of coefficients is the solution of the following linear least squares problem

$$\min \|^{\mathbb{p}}\mathbf{V}^k \boldsymbol{c} - \boldsymbol{y}\|,$$

where $\boldsymbol{y} \in \mathbb{R}^{N^Q}$ is the vector whose entries are the evaluation of $p$ at the internal quadrature points $\{\boldsymbol{x}_i^Q\}_{i=1}^{N^Q}$.

We will also analyze the condition number of the main local matrices that allow assembling the local system matrix, namely $\mathbf{G}^k$, $\mathbf{W}$, $\mathbf{B}$, $\mathbf{\Pi_k^0}$ and $\mathbf{D}$. The condition number of a matrix is computed as the ratio between its largest and its smallest singular values.

For the first two numerical examples, we consider problem (5) on the unit square domain $\Omega = (0, 1)^2$ with

$$\boldsymbol{\kappa} = \begin{bmatrix} y^2 + 1 & -xy \\ -xy & x^2 + 1 \end{bmatrix}, \quad \boldsymbol{b} = \begin{bmatrix} x \\ y \end{bmatrix}, \quad \gamma = x^2 + y^3.$$

The forcing term, the Dirichlet and the Neumann boundary conditions are set in such a way the exact solution is

$$p(x, y) = x^2 y + \sin(2\pi x) \sin(2\pi y) + 2, \tag{62}$$

and the set $\Gamma_N$ coincides with the edge of the domain of interest on the $x$-axis.

In particular, in the first experiment (Test1), we validate the proposed methods showing the order of convergence of the three monomial, partial and full orthogonal approaches against rising polynomial degree $k$, for various mesh refinement levels, on meshes made of squared elements. Then, in the second test (Test2), we show the performance of the approaches in terms of matrix condition number and error convergence trends in presence of collapsing polygons in the meshes.

Finally, the last example (Test3) proposes the application of the method to flow simulations in Discrete Fracture Networks (DFNs). DFN simulations are, indeed, a typical example where highly distorted mesh elements might appear in the process to generate a conforming mesh [8]. A simple network is here considered, for which an analytic solution is known, in order to compare the convergence curves of the three different approaches.

### 6.1 | Test1: Convergence rates

In this first experiment, we validate the new approaches by showing that the computational orders of convergence match the theoretical ones for different values of the polynomial degree $k$ on regular meshes. At this aim, convergence tests are performed on a sequence of squared meshes that decompose the domain of interest in 25, 100, 400 and 1600 identical squares, respectively.

Obtained convergence rates are shown in Table 1, while convergence curves of the errors are shown in Figure 1. In this figure, each row reports the graphs of the three error norms at varying $k$ on a fixed mesh. The three rows correspond to the last three considered refinement levels, that is, the 100, 400 and 1600 element meshes, respectively. In the figure, the upper bound of $y$-axis is fixed to 1.0e + 1.

The obtained results show that the rates of convergence related to all three approaches match the theoretical ones up to polynomial degree $k = 5$. For higher orders, errors in the monomial approach start to increase due to ill-conditioning, while both the partial and the full orthonormal approaches still provide the expected results in terms of errors, up to stagnation due to finite precision arithmetic. We highlight that, in the partial and full orthonormal approach, the local projection matrices and the global system matrix are well-conditioned, as we will see in the next tests.

### 6.2 | Test2: Collapsing polygons

Now, we analyze the condition number of local matrices in case of collapsing polygons. For this purpose, we use three meshes of rectangular elements, with elements having aspect ratios of 10, 50, and 100, respectively. We remark that the aspect ratio of an element is here defined as the ratio between the maximum and the minimum length of its edges. These meshes are built starting from a mesh made up of 100 identical squares, then subdivided into rectangles with length fixed to the value 0.1, equal to that of the original square, and a height computed according to the desired aspect ratio value. The mesh with aspect ratio of 10 originated from this process is shown in Figure 2.

Figures 3, 4 and 5 report the highest value among mesh elements of the condition number of elemental matrices $\mathbf{G}^k$, $\mathbf{W}$, $\mathbf{B}$, $\mathbf{\Pi_0^k}$ and $\mathbf{D}$, on the three considered meshes. In these figures, we set the upper bound of $y$-axis to 1.0e + 20. As shown in these figures, mass matrix $\mathbf{G}^k$, resulting from the full orthonormal approach, is perfectly well-conditioned, since it corresponds to the identity matrix, up to machine precision. Moreover, an algebraic growth of the condition number of the mass matrix obtained with the partial orthonormal approach is observed, whereas conditioning grows exponentially for the monomial one. More generally, we can observe that the condition numbers of the local matrices vary in a very limited range in the partial and full orthonormal approaches, as $k$ increases. Instead,

**TABLE 1** Test1: Convergence rates on squared mesh.

| | $k$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
|---|---|---|---|---|---|---|---|---|---|---|
| **Monomial** | $p_{\text{err}}$ | 1.1956 | 2.4760 | 3.7464 | 4.8565 | 5.9153 | 6.9456 | 5.8756 | 2.4067 | 1.5749 |
| | $u_{\text{err}}$ | 0.9947 | 2.0937 | 3.1263 | 4.0877 | 5.1110 | 5.9812 | 3.2093 | 0.3267 | -0.0167 |
| | $p_{I,\text{err}}$ | 1.9417 | 3.0056 | 3.9580 | 4.9588 | 5.9529 | 6.9680 | 5.8761 | 2.4071 | 1.5749 |
| **Partial** | $p_{\text{err}}$ | 1.1956 | 2.4760 | 3.7464 | 4.8565 | 5.9153 | 6.9456 | 7.6064 | 6.8582 | 5.1747 |
| | $u_{\text{err}}$ | 0.9947 | 2.0937 | 3.1263 | 4.0877 | 5.1110 | 6.0812 | 5.4221 | 3.8693 | 2.1663 |
| | $p_{I,\text{err}}$ | 1.9417 | 3.0056 | 3.9580 | 4.9588 | 5.9529 | 6.9680 | 7.6098 | 6.8590 | 5.1751 |
| **Ortho** | $p_{\text{err}}$ | 1.1956 | 2.4760 | 3.7464 | 4.8565 | 5.9153 | 6.9456 | 7.9637 | 8.5476 | 7.2031 |
| | $u_{\text{err}}$ | 0.9947 | 2.0937 | 3.1263 | 4.0877 | 5.1110 | 6.1058 | 7.0385 | 6.0692 | 4.3635 |
| | $p_{I,\text{err}}$ | 1.9417 | 3.0056 | 3.9580 | 4.9588 | 5.9529 | 6.9680 | 7.9773 | 8.5484 | 7.2035 |

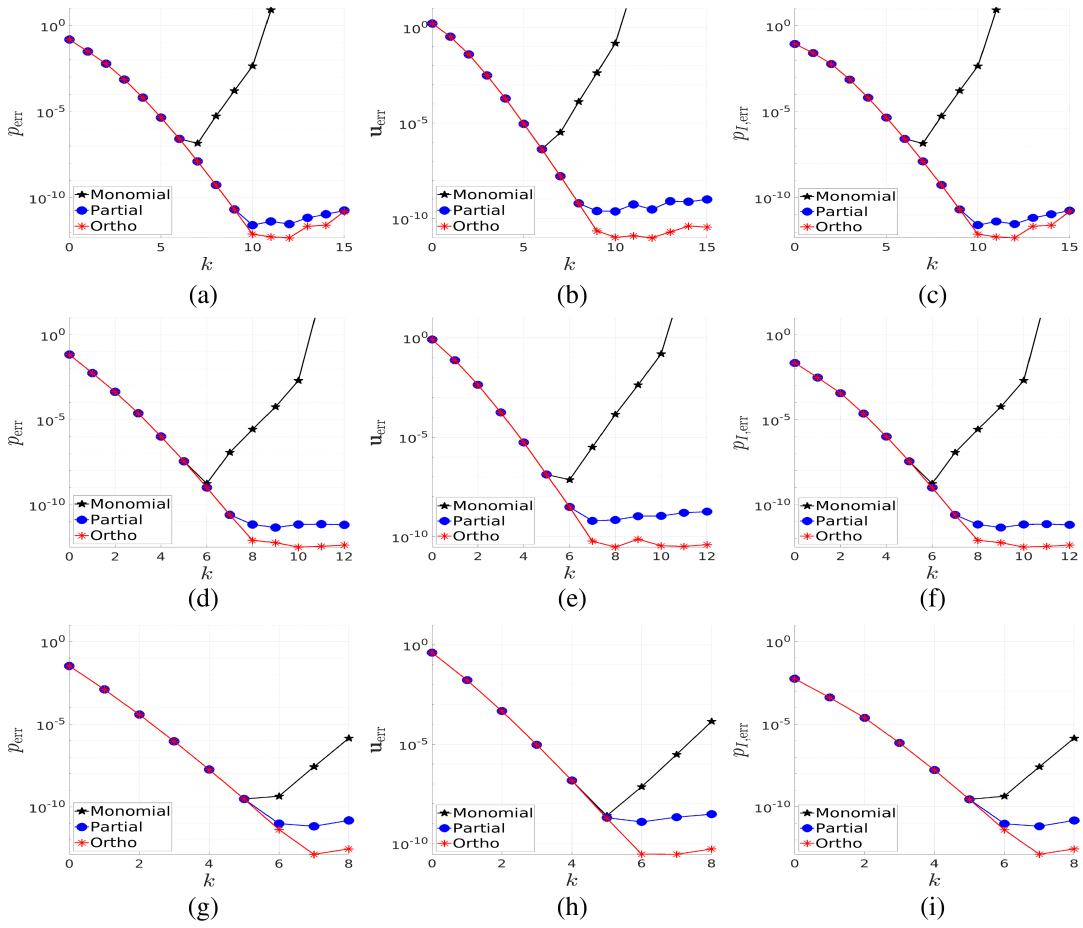


**FIGURE 1** Test1: Behaviour of errors (59), (60) and (61) at varying $k$ on square meshes. Pictures on each row refer to a different mesh refinement level: 100, 400 and 1600 element meshes from top to bottom.

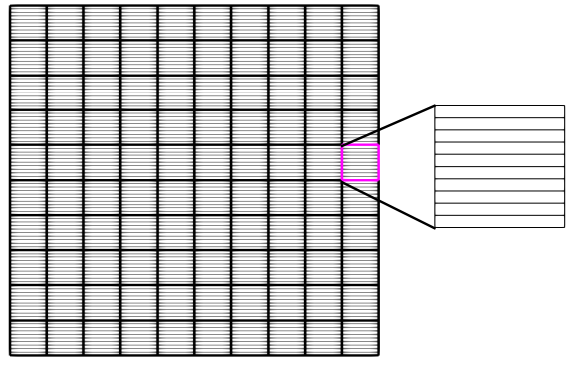

**FIGURE 2** Test2: Mesh with aspect ratio 10.

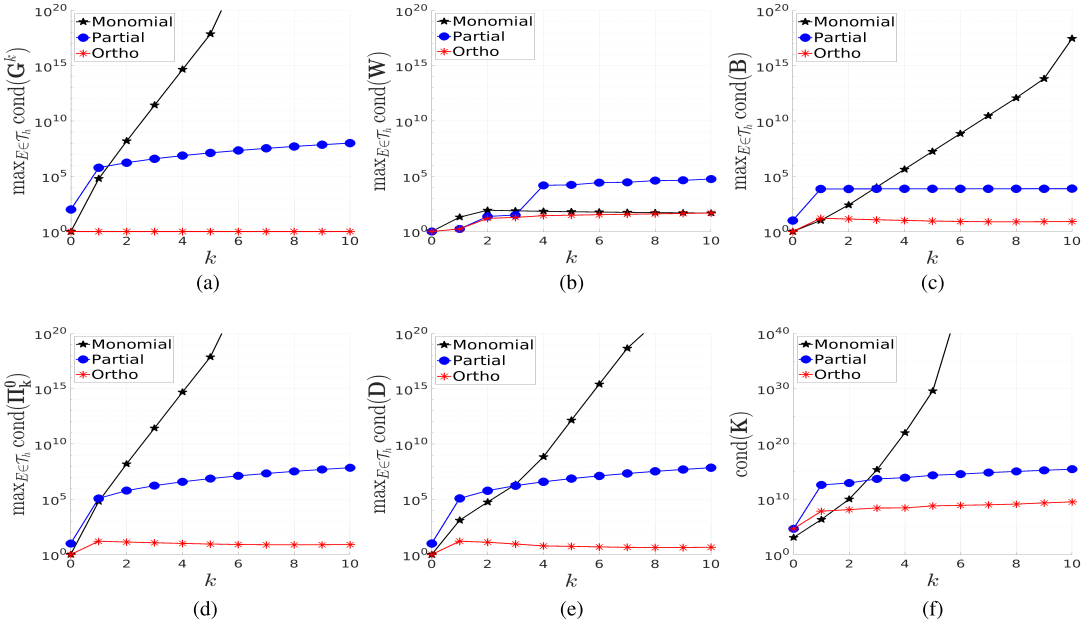


**FIGURE 3** Test2: Figure 3a–e show the maximum condition number of local matrices among elements, at varying $k$. Figure 3f reports the behaviour of the condition number of the global system matrix $\mathbf{K}$, at varying $k$. Mesh with aspect ratio 10.

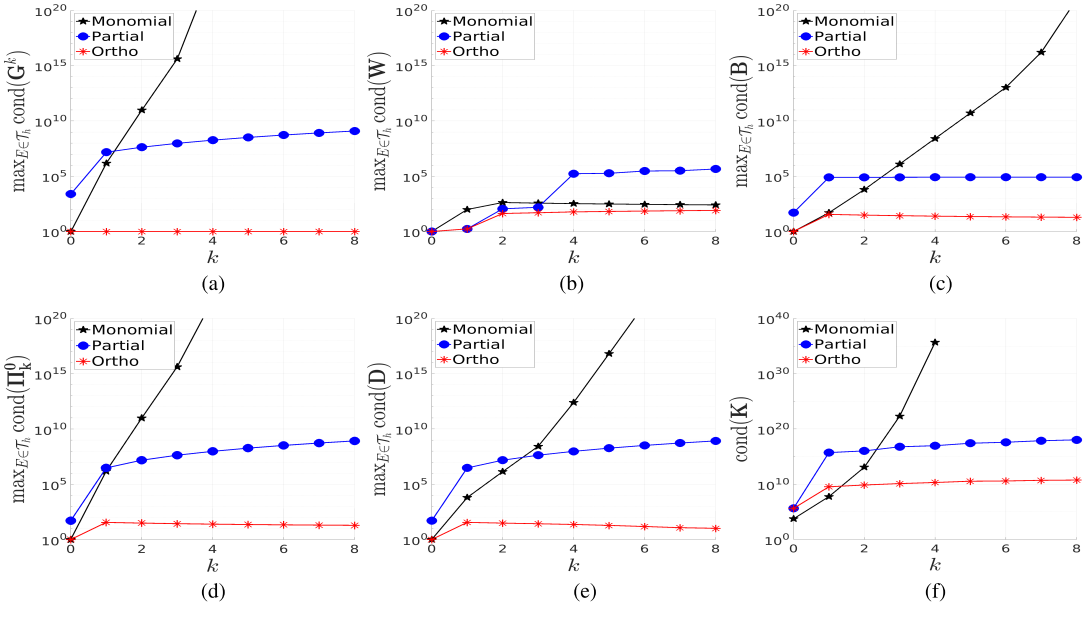


**FIGURE 4** Test2: Figure 4a–e show the maximum condition number of local matrices among elements, at varying $k$. Figure 4f reports the behaviour of the condition number of the global system matrix $\mathbf{K}$, at varying $k$. Mesh with aspect ratio 50.

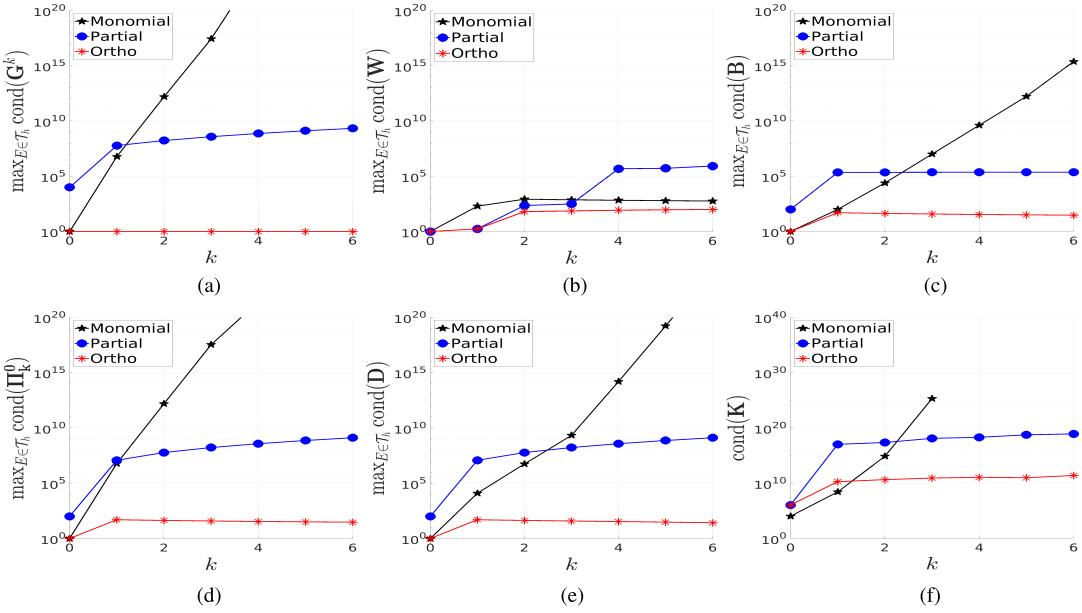


**FIGURE 5** Test2: Figure 5a–e show the maximum condition number of local matrices among elements, at varying $k$. Figure 5f reports the behaviour of the condition number of the global system matrix $\mathbf{K}$, at varying $k$. Mesh with aspect ratio 100.

the condition number of local matrices grows exponentially in the monomial approach, with the only exception of matrices $\mathbf{W}$, which exhibit a good conditioning regardless of the case.

Conditioning of matrices obtained with the partial-orthonormal bases appear to be slightly more affected by increasing aspect ratio values than the corresponding matrices with the full-orthonormal approach, however, still showing much lower values than the matrices given by the monomial basis.

Figure 6 reports convergence curves of the error against growing polynomial accuracy $k$, for the three considered meshes: top row for the mesh with aspect ratio 10, middle row for aspect ratio 50 and bottom row for aspect ratio 100. Also in this case, the upper bound of $y$-axis is fixed to 1.0e + 1. At low values of $k$, the curves corresponding to the monomial approach are well overlapped with the curves of partial and full-orthonormal approaches. Furthermore, the maximum value of polynomial accuracy $k$ for which the monomial approach provides errors in line with the other approaches reduces as the aspect ratio of mesh elements increases. Finally, error curves relating to the monomial approach are interrupted at values of $k \leq 6$ due to failure of linear algebra libraries (the *SparseLU* solver of Eigen) in computing a solution due to ill-conditioning of the global system matrix. Small differences are instead noticed between the curves obtained with the partial and full orthonormal approaches for all the considered values of $k$. We remark that these behaviours are coherent with the corresponding trends of the condition number of the global system matrices, which are shown in Figures 3f, 4f and 5f for the three considered meshes. In these figures, the upper bound of $y$-axis is set to 1.0e + 40. In the monomial approach, the condition number of the global system matrix $\mathbf{K}$ grows exponentially as the polynomial degree $k$ increases, while the growth in the partial and full orthonormal approaches is linear. The partial-orthonormal approach however provides condition numbers significantly higher than the full approach. In Figures 4f and 5f, the curves reporting the behaviour of the global condition numbers for the monomial approach are interrupted due to the failure of both the MATLAB R2023b routine *condest* and the C++ SuiteSparse v7.7.0 *klu_condest*. This behaviour confirms that using orthonormal

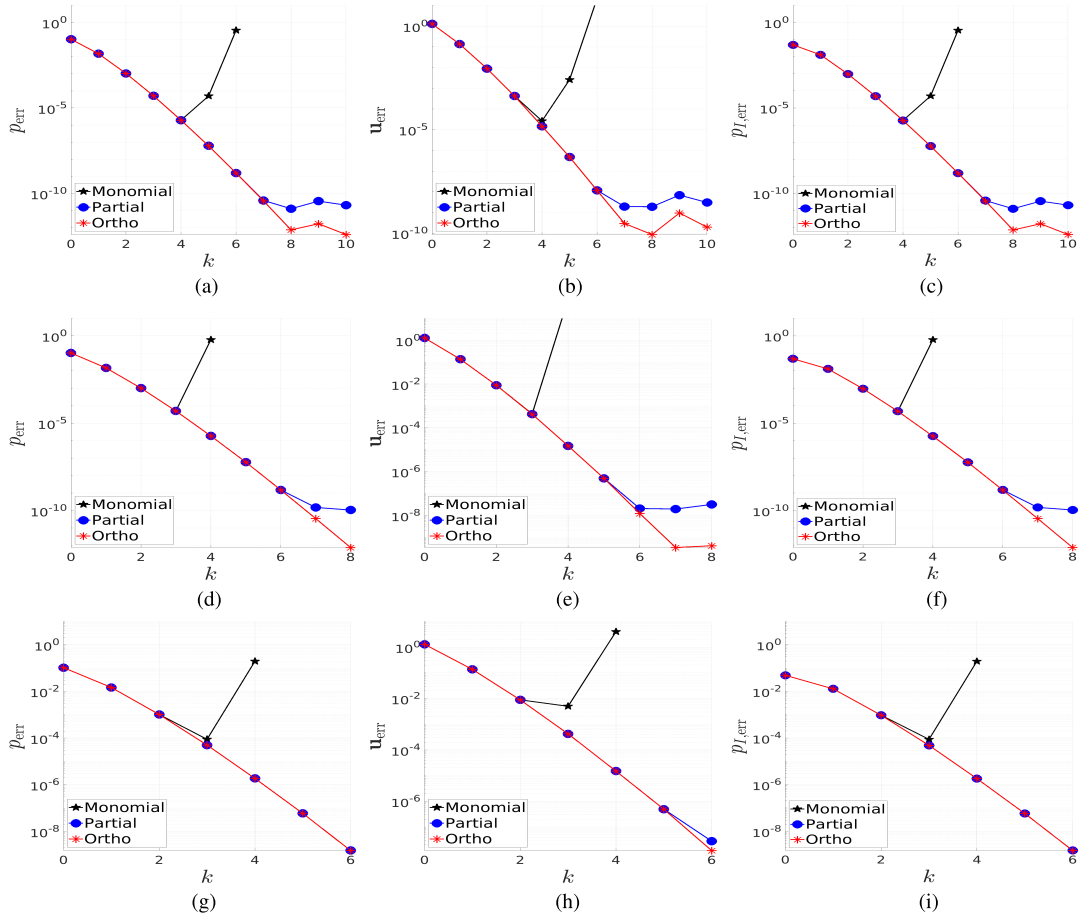


**FIGURE 6** Test2: Behaviour of errors (59), (60) and (61) at varying $k$ on rectangular meshes. Each row represents a different mesh: 10, 50, 100 from top to bottom.

polynomial basis to define the internal degrees of freedom allows to improve the conditioning of the global system matrix and to obtain more accurate solutions, as noted also in [15].

### 6.3 | Test3: Simulations in discrete fracture networks

In this last example, the application of the proposed orthonormal bases to Discrete Fracture Network (DFN) problems is presented. Discrete Fracture Networks are obtained as the union of planar polygonal domains with arbitrary orientations in the 3D space and are used to model the fractures in a porous medium [1]. As fracture thickness is typically orders of magnitude smaller than the other dimensions, fractures are geometrically reduced to 2D domains, and suitable equations, averaged across fracture thickness, are derived to describe the phenomena occurring in such domains [14]. Interface equations are then added at fracture intersections. A major complexity in DFN simulations consists in the generation of a conforming mesh, for realistic configurations characterized by intricate networks with a large number of fractures and fracture intersections. A possible strategy for DFN meshing is proposed, for example, in [8], based on the use of mixed virtual elements: first a triangular mesh is constructed on each fracture domain, independently of the intersections with the other fractures; then, the elements of these meshes are cut according to the interfaces, and hanging nodes are added where needed, to obtain

a fully conforming mesh of the whole domain (see [8] for more details). Highly elongated elements are likely to be generated in this process, such that the standard choice of VEM basis function might yield badly conditioned problems for high order approximations [12].

An advection-diffusion-reaction problem on a network made up by the union of three fractures $F_i$ with three intersections $\Gamma_i$, is here considered, namely

$$
\begin{aligned}
F_1 &= \{(x,y,z) \in \mathbb{R}^3 : -1 \le x \le 1, -1 \le y \le 1, z = 0\},\\
F_2 &= \{(x,y,z) \in \mathbb{R}^3 : -1 \le z \le 1, -1 \le x \le 1, y = 0\},\\
F_3 &= \{(x,y,z) \in \mathbb{R}^3 : -1 \le y \le 1, -1 \le z \le 1, x = 0\},
\end{aligned}
$$

$$
\begin{aligned}
\Gamma_1 &= \{(x,y,z) \in \mathbb{R}^3 : -1 \le x \le 1, y = 0, z = 0\},\\
\Gamma_2 &= \{(x,y,z) \in \mathbb{R}^3 : -1 \le y \le 1, x = 0, z = 0\},\\
\Gamma_3 &= \{(x,y,z) \in \mathbb{R}^3 : -1 \le z \le 1, x = 0, y = 0\}.
\end{aligned}
$$

On each fracture, we choose

$$
\boldsymbol{\kappa}_i(\hat{x},\hat{y}) = \begin{bmatrix} 1+\hat{y}^2 & -\frac{\hat{x}\hat{y}}{2} \\ -\frac{\hat{x}\hat{y}}{2} & 1+\hat{x}^2 \end{bmatrix}, \quad \boldsymbol{b}_i(\hat{x},\hat{y}) = \begin{bmatrix} \hat{x}-\hat{y} \\ \hat{y}-1 \end{bmatrix}, \quad \gamma_i(\hat{x},\hat{y}) = \hat{x}^3 + \hat{y}.
$$

where $(\hat{x},\hat{y})$ is a proper fracture-local reference system, and a forcing term and Neumann boundary conditions are defined in such a way the exact solution on each fracture is:

$$
\begin{aligned}
h_1(x,y) &= -|x|(1+x)(1-x)y(1+y)(1-y),\\
h_2(z,x) &= -z(1+z)(1-z)x(1+x)(1-x),\\
h_3(y,z) &= y(1+y)(1-y)|z|(1+z)(1-z).
\end{aligned}
$$

The same problem is considered in [8] up to order 5. The network and the exact solution are shown in Figure 7.

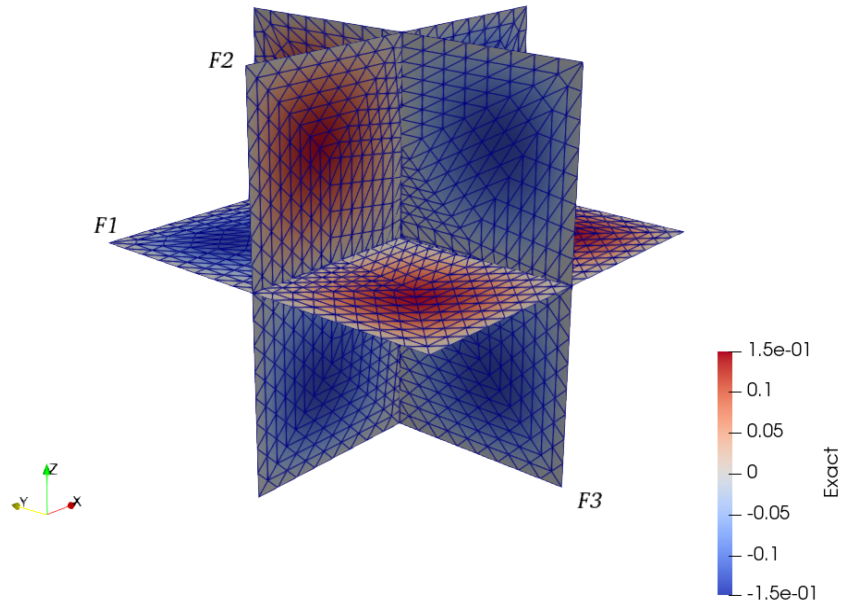


FIGURE 7 Test3: Exact solution DFN benchmark problem.

TABLE 2 Test3: Number of cells (Num Cells), minimum aspect ratio (Min AR) and maximum aspect ratio (Max AR) of mesh cells for the four refinement levels (R0 to R3).

| | Num cells | Min AR | Max AR |
|---|---|---|---|
| **R0** | 75 | 1.41 | 38.43 |
| **R1** | 246 | 1.19 | 62.11 |
| **R2** | 882 | 1.19 | 152.11 |
| **R3** | 3294 | 1.14 | 215.13 |

TABLE 3 Test3: Convergence rates on conforming mesh.

| | $k$ | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|---|
| **Monomial** | $p_{\text{err}}$ | 1.1984 | 2.3292 | 3.5494 | 4.6377 | 5.6597 | 2.8423 | 1.8227 | −4.7912 |
| | $u_{\text{err}}$ | 1.0354 | 1.8430 | 2.8787 | 3.9900 | 4.9328 | 1.1135 | 0.6710 | −6.5224 |
| | $p_{I,\text{err}}$ | 1.7581 | 2.6572 | 3.8542 | 4.8566 | 5.8788 | 2.8333 | 1.8227 | −4.7912 |
| **Partial** | $p_{\text{err}}$ | 1.1984 | 2.3292 | 3.5494 | 4.6377 | 5.6633 | 6.8989 | 5.7860 | −2.8821 |
| | $u_{\text{err}}$ | 1.0354 | 1.8430 | 2.8787 | 3.9900 | 5.0341 | 4.7002 | 2.8836 | −4.0248 |
| | $p_{I,\text{err}}$ | 1.7581 | 2.6572 | 3.8542 | 4.8566 | 5.8885 | 6.9844 | 5.7860 | −2.8821 |
| **Ortho** | $p_{\text{err}}$ | 1.1984 | 2.3292 | 3.5494 | 4.6377 | 5.6633 | 6.9237 | 8.0005 | −0.3784 |
| | $u_{\text{err}}$ | 1.0354 | 1.8430 | 2.8787 | 3.9900 | 5.0345 | 6.0422 | 6.2590 | −2.7340 |
| | $p_{I,\text{err}}$ | 1.7581 | 2.6572 | 3.8542 | 4.8566 | 5.8885 | 7.0225 | 8.0005 | −0.3784 |

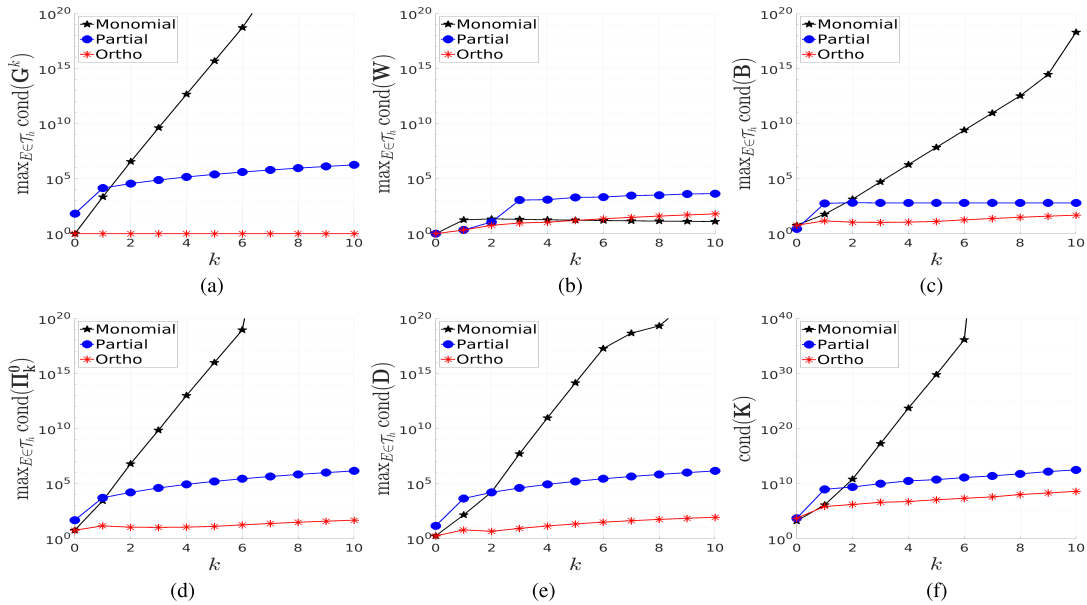


FIGURE 8 Test3: Figure 8a–e show the maximum condition number of local matrices among elements, at varying $k$. Figure 8f reports the behaviour of the condition number of the global system matrix $\mathbf{K}$, at varying $k$. Coarsest mesh.

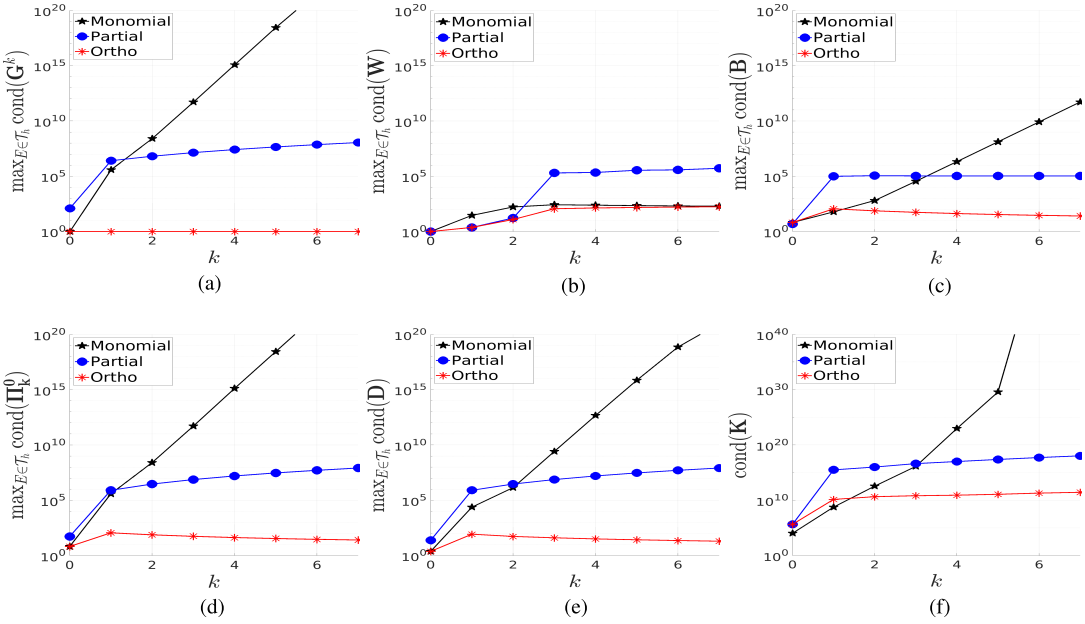


**FIGURE 9** Test3: Figure 9a–e show the maximum condition number of local matrices among elements, at varying $k$. Figure 9f reports the behaviour of the condition number of the global system matrix $\mathbf{K}$, at varying $k$. Finest mesh.

In this numerical test, we use four refinements of an initially triangular mesh, modified, as mentioned above, in such a way that the final polygonal meshes are conforming at the traces. Table 2 reports the number, the minimum and maximum aspect ratio of mesh elements for the four refinements (R0 to R3).

Table 3 shows the computed convergence rates of errors (59) and (61), for all the three tested approaches. In each sub-domain, the velocity field is a vector of polynomials of degree 7, such that, for $k \geq 7$ only errors related to floating point arithmetic computations are to be expected.

Figures 8 and 9 report, as previously, the maximum condition number across mesh elements of the computed local matrices and the trend of the condition number of the global system matrix as $k$ varies, on the coarsest and finest considered meshes. We can observe that these data show the same behaviour as in the previous tests. Figure 10 shows error convergence curves against polynomial accuracy $k$, for the four considered mesh refinement levels. Pictures on each row correspond to the same mesh. It can be seen that the curves obtained with the three approaches are almost indistinguishable for $k$ up to $4-5$. For higher values errors given by the monomial approach start growing rapidly due to ill-conditioning, whereas errors given by partial and full orthonormal approaches still decrease up to stagnation due to finite precision arithmetic.

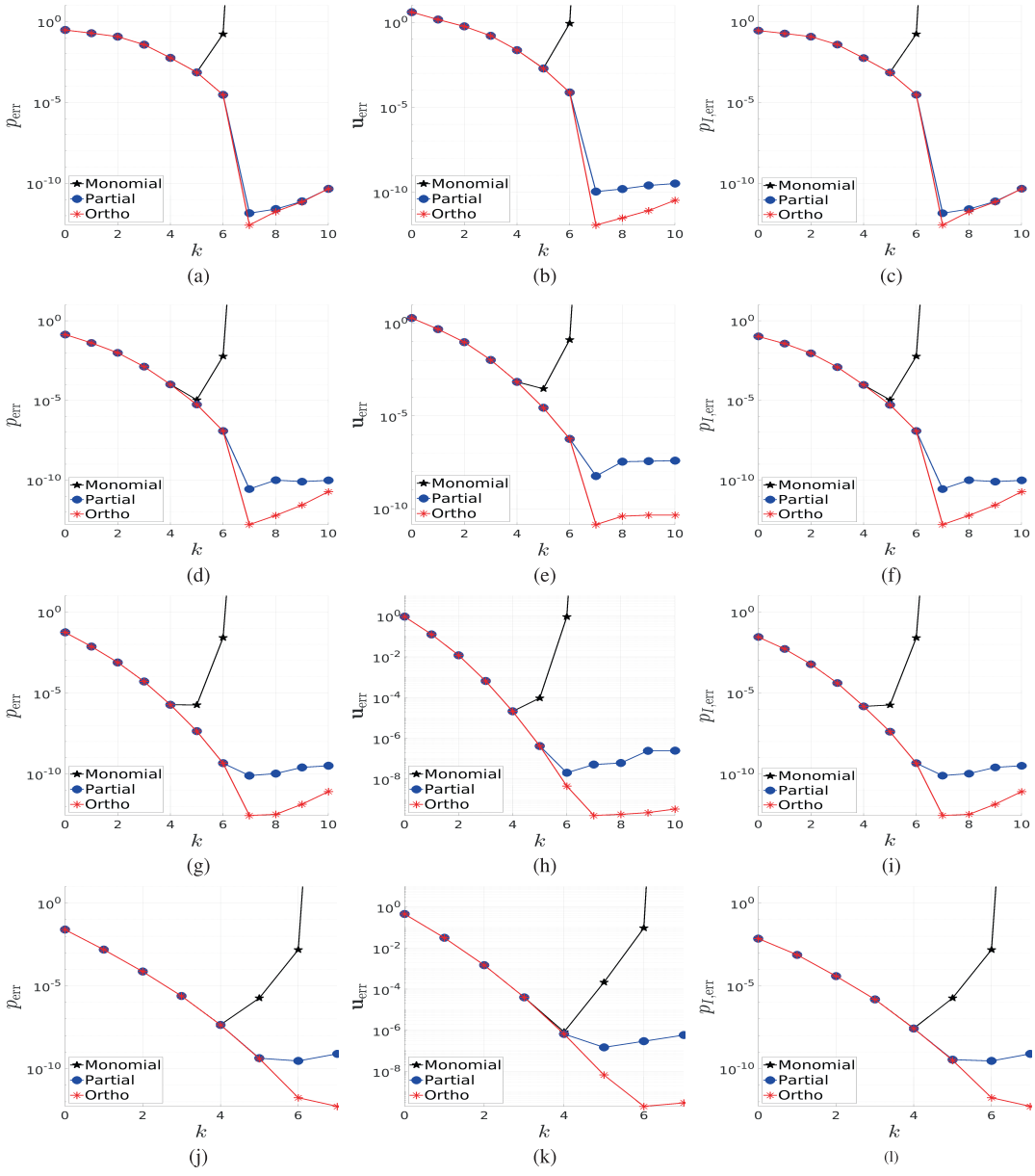


**FIGURE 10** Test3: Behaviour of errors (59), (60) and (61) at varying $k$ on conforming meshes. Each row represents a different refinement, from the coarsest mesh on top to the finest mesh at the bottom.

## 7 | CONCLUSIONS

In this paper, we presented a possible solution to cure the ill-conditioning of system matrix in the mixed formulation of the Virtual Element Method.

Since in the mixed formulation we need to introduce a discrete local space for both the pressure and the velocity variable, we have first introduced an orthonormal scalar-polynomial basis in the pressure space and then we have also orthonormalized the vector-polynomial basis used in the definition of the degrees of freedom related to the velocity variable.

Numerical experiments suggest that the introduction of orthonormal polynomial basis in both spaces allows to improve stability of mixed Virtual Elements for high order applications on distorted elements.

The additional computational cost for the Ortho and the Partial approaches with respect to the standard, monomial approach stems from the application of the Gram-Schmidt algorithm. This needs to be performed twice for the Partial approach and three times for the full-orthonormal approach. However, its cost is only associated with local quantities, such as the local polynomial degree $k$ and the cardinality of the employed quadrature formula. Additionally, in the full-orthonormal approach, this cost is at least partially mitigated by the elimination of the two linear system resolutions, namely (58) and (55). It is also to remark that, overall, the leading cost is the one related to the resolution of the global system, that does not increase after the use of the proposed polynomial bases, and may even decrease due to improvements in the condition number.

It is worth to mention that the methods here suggested to build orthonormal polynomial bases improve the conditioning of the Vandermonde matrix defined with respect to the quadrature formulas in the interior of each element $E$. However, in general, this does not guarantee an improvement in the conditioning of the Vandermonde matrix defined with respect to quadrature formulas on the boundary of the elements [10]. Nonetheless, this appears to be sufficient to recover optimal convergence trends in the considered cases.

## ACKNOWLEDGMENTS

The author S.B. kindly acknowledges partial financial support provided by PRIN project "Advanced polyhedral discretisations of heterogeneous PDEs for multiphysics problems" (No. 20204LN5N5_003) and by PNRR M4C2 project of CN00000013 National Centre for HPC, Big Data and Quantum Computing (HPC) (CUP: E13C22000990001). The author S.S. kindly acknowledges partial financial support provided by INdAM-GNCS through project "Sviluppo ed analisi di Metodi agli Elementi Virtuali per processi accoppiati su geometrie complesse" and that this publication is part of the project NODES which has received funding from the MUR-M4C2 1.5 of PNRR with grant agreement no. ECS00000036. The author G.T. kindly acknowledges financial support provided by the MIUR programme "Programma Operativo Nazionale Ricerca e Innovazione 2014–2020" (CUP: E11B21006490005). Computational resources are partially supported by SmartData@polito. Open access publishing facilitated by Politecnico di Torino, as part of the Wiley - CRUI-CARE agreement.

## DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

## ORCID

*Gioana Teora* https://orcid.org/0000-0002-8540-3639

## REFERENCES

[1] P. M. Adler, J. F. Thovert, and V. V. Mourzenko, *Fractured porous media*, Oxford University Press, Oxford, UK, 2012.

[2] F. Bassi, L. Botti, A. Colombo, D. Di Pietro, and P. Tesini, *On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations*, J. Comput. Phys. 231 (2012), no. 1, 45–65. https://doi.org/10.1016/j.jcp.2011.08.018.

[3] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, and A. Russo, *Basic principles of virtual element methods*, Math Models Methods Appl Sci 23 (2013), no. 1, 199–214. https://doi.org/10.1142/S0218202512500492.
[4] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo, *The Hitchhiker's guide to the virtual element method*, Math Models Methods Appl Sci 24 (2014), no. 8, 1541–1573. https://doi.org/10.1142/S021820251440003X.
[5] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo. H(div) and H(curl)-conforming VEM. 2016a. https://doi.org/10.1007/s00211-015-0746-1.
[6] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo, *Mixed virtual element methods for general second order elliptic problems on polygonal meshes*, Esaim: M2AN 50 (2016b), no. 3, 727–747. https://doi.org/10.1051/m2an/2015067.
[7] L. Beirão da Veiga, F. Brezzi, L. D. Marini, and A. Russo, *Virtual element implementation for general elliptic equations*, Springer International Publishing, Cham, 2016c, 39–71. https://doi.org/10.1007/978-3-319-41640-3_2.
[8] M. F. Benedetto, A. Borio, and S. Scialò, *Mixed virtual elements for discrete fracture network simulations*, Finite Elem. Anal. Des. 134 (2017), 55–67. https://doi.org/10.1016/j.finel.2017.05.011.
[9] S. Berrone and A. Borio, *Orthogonal polynomials in badly shaped polygonal elements for the virtual element method*, Finite Elem. Anal. Des. 129 (2017), 14–31. https://doi.org/10.1016/j.finel.2017.01.006.
[10] S. Berrone, S. Scialò, and G. Teora, *The mixed virtual element discretization for highly-anisotropic problems: The role of the boundary degrees of freedom*, Math Eng 5 (2023), no. 6, 1–32. https://doi.org/10.3934/mine.2023099.
[11] F. Brezzi, R. S. Falk, and L. Donatella Marini, *Basic principles of mixed virtual element methods. ESAIM*, Math Model Numer Anal 48 (2014), no. 4, 1227–1240. https://doi.org/10.1051/m2an/2013138.
[12] F. Dassi, A. Fumagalli, D. Losapio, S. Scialò, A. Scotti, and G. Vacca. The Mixed Virtual Element Method for Grids with Curved Interfaces in Single-Phase Flow Problems. 2021. https://doi.org/10.2118/203998-MS.
[13] L. Giraud, J. Langou, and M. Rozloznik, *The loss of orthogonality in the gram-Schmidt orthogonalization process*, Comput Math Appl 50 (2005), no. 7, 1069–1075. https://doi.org/10.1016/j.camwa.2005.08.009.
[14] V. Martin, J. Jaffré, and J. E. Roberts, *Modeling fractures and barriers as interfaces for flow in porous media*, SIAM J. Sci. Comput. 26 (2005), no. 5, 1667–1691. https://doi.org/10.1137/S1064827503429363.
[15] L. Mascotto, *Ill-conditioning in the virtual element method: Stabilizations and bases*, Numer Methods Partial Differ Equ 34 (2018), no. 4, 1258–1281. https://doi.org/10.1002/num.22257.

**How to cite this article:** S. Berrone, S. Scialò, and G. Teora, *Orthogonal polynomial bases in the mixed virtual element method*, Numer. Methods Partial Differ. Eq. (2024), e23144. https://doi.org/10.1002/num.23144