Orthogonal polynomials in badly shaped polygonal elements for the Virtual Element Method

Orthogonal polynomials in badly shaped polygonal elements for the Virtual Element Method[☆]

S. Berrone[a,*], A. Borio[a]

[a] *Dipartimento di Scienze Matematiche, Politecnico di Torino*
Corso Duca degli Abruzzi 24, Torino, 10129, Italy

Abstract

In this paper we propose a modified construction for the polynomial basis on polygons used in the Virtual Element Method (VEM). This construction is alternative to the classical monomial basis used in the classical construction of the VEM and is designed in order to improve numerical stability. For badly shaped elements the construction of the projection matrices required for assembling the local coefficients of the linear system within the VEM discretization of Partial Differential Equations can result very ill conditioned. The proposed approach can be easily implemented within an existing VEM code in order to reduce the possible ill conditioning of the elemental projection matrices. Numerical results applied to an hydro-geological flow simulation that often produce very badly shaped elements show a clear improvement of the quality of the numerical solution, confirming the viability of the approach. The method can be conveniently combined with a classical implementation of the VEM and applied element-wise, thus requiring a rather moderate additional numerical cost.

Keywords: VEM, polygonal Galerkin methods, orthogonal polynomials on polygons, Discrete Fracture Network simulations, badly shaped elements.
2010 MSC: 65N30, 65N50, 68U20, 86-08, 86A05

1. Introduction

In the recent years a large interest on polythopal methods for PDEs has rapidly grown. In many fields of computational engineering and scientific computing the geometrical complexity is often as relevant as the model complexity. In all these situations the introduction of polyhedral or polygonal methods can

[☆]This research has been partially supported by the Italian MIUR through PRIN research grant 2012HBLYE4_001 *Metodologie innovative nella modellistica differenziale numerica* and by INdAM-GNCS. Computational resources were partially provided by HPC@POLITO (http://hpc.polito.it).

[*]Corresponding author
Email addresses: stefano.berrone@polito.it (S. Berrone), andrea.borio@polito.it (A. Borio)

introduce a large and useful flexibility that can play a relevant role in simulations.

This improved flexibility of the recently developed Virtual Element Method (VEM) has been applied in the field of geological poro-fractured media [1–5]. Geosciences very often produce applications with huge domains and terrific geometrical complexities. Within this context, the Discrete Fracture Network (DFN) model was developed for modeling the flow in the geological fractured media [6–9] and is object of a very large numerical bibliography [10–20]. Due to the huge uncertainty in the definition of the underground fracture distribution, this model instantiates a fracture distribution by a stochastic procedure starting from probabilistic distributions of geometrical parameters: direction, dimension, aspect ratio; and from probabilistic distributions of thickness and other hydrogeological properties. The stochastic procedure that instantiates the fracture distribution can create geometrical complexities arbitrarily demanding for a numerical method; typically these complexities are related, for example, to very small angles between couple of fractures, to a huge variability in the length of fracture-intersections, and to disjoint fractures very close to each other [21]. Several approaches were recently applied to the DFN flow problem [10–14]. In some of these methods some geometrical simplification were required in order to construct the mesh. In [1, 2, 21–28] an optimization approach was developed in order to overcome these geometrical complexities by-passing the constraints imposed on the mesh generation process. This optimization-based approach was applied in conjunction to the classical Finite Element Method (FEM) as well as with the eXtended Finite Element Method and the VEM [1]. The VEM applied to this problem has proved a good reliability in dealing with these complexities, but, sometimes, some fracture configurations have lead to unfeasible numerical solutions [2]. A possible solution, sometimes viable, is to relax the mesh conformity requirement, resorting to a Mortar fracture matching method [3] or applying a preliminary *mesh smoothing process* [3]. Nonetheless, some very badly shaped configurations cannot be avoided, mainly on coarse meshes.

The Virtual Element Method was recently developed as a generalization of Mimetic Finite Differences,[29, 30], and has been applied to a wide number of problems, such as plate bending problems [31], elasticity problems [32, 33], Stokes problems [34] and the Steklov eigenvalue problem [35].

Starting from these observations, in this paper we propose a different basis for assembling the local linear systems within the VEM, that, at a very small additional cost with respect to a classical implementation based on monomials, can largely improve the reliability of the method by limiting the condition numbers of local matrices in badly shaped elements. We remark that the proposed method aims at improving the reliability of the computations performed in the set up of the consistent part of the VEM formulation of the problem and is completely independent of the VEM stabilization that is added to the consistent part in order to get a well posed problem [36]. Moreover, our description is organized in such a way that it can be easily plugged in a standard VEM code based on scaled monomials.

In Section 3 we introduce the computation of a quasi-orthogonal polynomial

basis for assembling the VEM linear system that is fully compatible with the traditional monomial basis. The two bases can be mixed on elements in the same mesh using the quasi-orthogonal basis on badly shaped elements and the traditional monomial basis on all the other elements. In Section 4 we provide a brief validation of the modified VEM construction on a general reaction-convection-diffusion problem with variable coefficients. In Section 5 we compare the results provided by the classical monomial basis with the presented quasi-orthogonal basis on two critical Discrete Fracture Networks. In this Section we further discuss some simple criteria useful to determine in which elements it is beneficial to resort to the new basis and in which elements it is safe to use the monomial basis, as well as some limitations of the proposed approach.

2. Virtual Element Spaces

The Virtual Element Method [37, 38] is a recently developed Galerkin approach to PDEs that aims at allowing the use of more generally shaped polygons than the ones allowed in the FEM context.

Consider a bounded open set $\Omega \subset \mathbb{R}^2$, partitioned by a mesh $\mathcal{T}_h$ made up of open star-shaped polygons having an arbitrary finite number of sides (even different from one polygon to another). We make the following regularity assumption: $\exists \gamma > 0$ such that $\forall E \in \mathcal{T}_h$, with diameter h_E, E is star-shaped with respect to a ball of radius larger than γh_E; more details on the regularity assumptions can be found in [36].

We define $\Pi_k^\nabla : \mathrm{H}_0^1(\Omega) \to \mathbb{P}_k(\mathcal{T}_h)$ such that, $\forall v \in \mathrm{H}_0^1(\Omega)$ and $\forall E \in \mathcal{T}_h$

$$\left(\nabla\left(v - \Pi_k^\nabla v\right), \nabla p\right)_E = 0\,, \forall p \in \mathbb{P}_k(E) \quad \text{and} \quad \begin{cases} \left(\Pi_k^\nabla v, 1\right)_{\partial E} = (v,1)_{\partial E} & \text{if } k = 1\,, \\ \left(\Pi_k^\nabla v, 1\right)_E = (v,1)_E & \text{if } k \geq 1\,, \end{cases}$$

where, $\forall E \in \mathcal{T}_h$, $\mathbb{P}_k(E)$ is the space of polynomials of degree up to k; its dimension is $\dim\left(\mathbb{P}_k(E)\right) = \frac{(k+1)(k+2)}{2}$.

Following [38, 39], we introduce the local finite dimensional space

$$V_h^E := \left\{ v \in \mathrm{H}^1(E) : \Delta v \in \mathbb{P}_k(E)\,,\ v \in \mathbb{P}_k(e)\ \forall e \subset \partial E, \right. \\ \left. (v,p)_E = \left(\Pi_k^\nabla v, p\right)_E\ \forall p \in \mathbb{P}_k(E)/\mathbb{P}_{k-2}(E) \right\},$$

where the space $\mathbb{P}_k(E)/\mathbb{P}_{k-2}(E)$ is intended to contain those polynomials belonging to $\mathbb{P}_k(E)$ that are $\mathrm{L}^2(E)$-orthogonal to the ones in $\mathbb{P}_{k-2}(E)$. We then define the global Virtual Element Space on $\mathcal{T}_h$ by gluing local spaces asking for continuity:

$$V_h := \left\{ v \in C^0(\Omega) \cap \mathrm{H}_0^1(\Omega) : v \in V_h^E\ \forall E \in \mathcal{T}_h \right\}.$$

The following degrees of freedom are unisolvent for V_h (see [37, 38]):

1. the values at the vertices of the polygon;
2. if $k \geq 2$, for each edge $e \subset \partial E$, the value of $v \in V_h$ at $k-1$ internal points of e. For practical purposes, we choose these points to be the internal Gauss – Lobatto quadrature nodes;

3. if $k \geq 2$, the scaled moments $\frac{1}{|E|}(v, m_{\boldsymbol{\alpha}})_E$, for all the scaled monomials $m_{\boldsymbol{\alpha}} \in \mathcal{M}_{k-2}(E)$ up to the order $k-2$, defined as

$$\forall \mathbf{x} = (x, y) \in E, \quad m_{\boldsymbol{\alpha}}(x, y) := \frac{(x - x_E)^{\alpha_1}(y - y_E)^{\alpha_2}}{h_E^{\alpha_1+\alpha_2}}, \tag{1}$$

with $\boldsymbol{\alpha} = (\alpha_1, \alpha_2)$, $|\boldsymbol{\alpha}| = \alpha_1 + \alpha_2 \leq k-2$.

The above degrees of freedom are enough to build projection matrices in order to obtain local polynomial orthogonal projections from V_h to $\mathbb{P}_k(\mathcal{T}_h)$, see [40].

2.1. Example: VEM for advection-diffusion-reaction equations

Following [38], we consider the general second order problem

$$\begin{cases} -\nabla \cdot (\mu \nabla u) + \beta \cdot \nabla u + \gamma u = f & \text{in } \Omega, \\ u = 0 & \text{on } \partial\Omega, \end{cases}$$

whose variational formulation reads

$$(\mu \nabla u, \nabla v) + (\beta \cdot \nabla u, v) + (\gamma u, v) = (f, v) \ . \tag{2}$$

The VEM discretization of (2) consists in defining a discrete counterpart of the bilinear form which is computable from the VEM degrees of freedom. Let

$$\begin{aligned} a_h(u_h, v_h) &:= \left(\mu \Pi^0_{k-1} \nabla u_h, \Pi^0_{k-1} \nabla v_h\right) + S\left(\left(I - \Pi^\nabla_k\right) u_h, \left(I - \Pi^\nabla_k\right) v_h\right), \\ b_h(u_h, v_h) &:= \left(\beta \cdot \Pi^0_{k-1} \nabla u_h, \Pi^0_{k-1} v_h\right), \\ c_h(u_h, v_h) &:= \left(\Pi^0_{k-1} u_h, \Pi^0_{k-1} v_h\right), \\ \mathcal{B}_h(u_h, v_h) &:= a_h(u_h, v_h) + b_h(u_h, v_h) + c_h(u_h, v_h), \end{aligned}$$

where S is the VEM stabilization [36, 37] such that

$$\exists c_*, c^* > 0 \colon \forall v_h \in \ker\left(\Pi^\nabla_k\right), \ c_* \left\|\nabla v_h\right\|^2 \leq S(v_h, v_h) \leq c^* \left\|\nabla v_h\right\|^2 ,$$

and all the other terms of the operator $\mathcal{B}_h(.,.)$ provide the consistent part of the operator. Within these terms, the operator Π^0_{k-1} is the elementwise $\mathrm{L}^2(E)$ projection on $\mathbb{P}_{k-1}(E)$, for any $E \in \mathcal{T}_h$. For the ease of notation, we will use the same symbol also for the application of the projection operator to vectors, such as gradients, meaning a component-wise application.

Using the above definitions, we define the discrete VEM solution as the function $u_h \in V_h$ satisfying

$$\mathcal{B}_h(u_h, v_h) = \left(f, \Pi^0_{k-1} v_h\right) \quad \forall v_h \in V_h \, .$$

This problem is well-posed and satisfies optimal a priori error estimates [38]. In the following we focus on the construction of the local projection matrices and the local matrices and vectors required for the set up of the global discrete problem.

In the presentation given here we have considered the minimal requirement in the projections in order to preserve the expected polynomial rate of convergence (k in the energy norm) of the numerical solution. In the first VEM papers the projection used for the right-hand-side computation was ${}^{\mathbf{m}}\boldsymbol{\Pi}^0_k$.

3. Orthogonal polynomials on the generic element

All the computations performed in order to set up the VEM linear system providing the solution are based on operations between polynomial functions representing the projection of functions appearing in the consistent part of the operator and in the right-hand-side. A key issue in performing all the computations is a suitable basis for the polynomial spaces on general polygonal elements. Among the several possible options the classical and more simple choice is the scaled monomial basis [37, 38]. In the following we describe the construction of a suitable different almost orthogonal basis. A key issue to be considered in this construction process is that we need a basis for the space of polynomials of order $k-1$ for the construction of the Π^0_{k-1} projector, largely used in the consistent part of the discretization of the problem. This is the first step of our construction. Moreover, we also need a basis for the full space of polynomials of order k for the computations involved by the Π^∇_k construction required in the VEM stabilization considered in [37, 38]. For this reason we need a basis for the space $\mathbb{P}_k(E)$ obtained by the chosen basis functions for $\mathbb{P}_{k-1}(E)$ and by a set of additional linearly independent basis functions. We remark that the proposed construction of a polynomial basis aims at improving the reliability of the projector operator and is not dependent on the VEM stabilization chosen [36].

3.1. Basis construction

Let $\mathbf{m}^k$ be the column vector of the n_k scaled monomial basis functions of the space of polynomials up to degree k usually used in the VEM definition, and $\mathbf{p}^k$ the column vector of a suitable set of linearly independent polynomials of degree k, whose construction will be discussed in the following.

Let $\mathbf{R}^k$ be the matrix whose i-th row represent the coefficients of the i-th polynomial p^k_i of the orthogonal basis in terms of the monomial basis $\mathbf{m}^k$:

$$p^k_i = \sum_{j=1,..n_k} r_{i,j} m^k_j = \mathbf{R}^k_{i,:}\mathbf{m}^k\,.$$

In a compact form we can write

$$\mathbf{p}^k = \mathbf{R}^k\mathbf{m}^k.$$

Let us introduce the mass matrix ${}^{\mathbf{m}}\mathbf{H}^k \in \mathbb{R}^{n_k\times n_k}$ defined as

$${}^{\mathbf{m}}\mathbf{H}^k = \int_E \mathbf{m}^k \left(\mathbf{m}^k\right)^T \mathrm{d}\Omega\,,$$

and let us consider the principal sub-matrix of order n_{k-1}, that is the mass matrix of the monomials up to the order $k-1$:

$${}^{\mathbf{m}}\mathbf{H}^{k-1} = \int_E \mathbf{m}^{k-1} \left(\mathbf{m}^{k-1}\right)^T \mathrm{d}\Omega\,.$$

Moreover, let us denote by ${}^{\mathbf{m}}\mathbf{H}^{k,k-1}$ the block of the mass matrix ${}^{\mathbf{m}}\mathbf{H}^k$ with the last $n_k - n_{k-1}$ rows and the first n_{k-1} columns, and by ${}^{\mathbf{m}}\mathbf{H}^{k/k-1}$ the block matrix with the last $n_k - n_{k-1}$ rows and columns.

Orthonormal basis for $\mathbb{P}_{k-1}(E)$. Let us define the matrix $\mathbf{R}^{k-1}$ such that the mass matrix ${}^{\mathbf{p}}\mathbf{H}^{k-1}$ with respect to the basis $\mathbf{p}^{k-1}$ is diagonal:

$$
\begin{aligned}
{}^{\mathbf{p}}\mathbf{H}^{k-1} = \int_E \mathbf{p}^{k-1} \left(\mathbf{p}^{k-1}\right)^T \mathrm{d}\Omega = \int_E \mathbf{R}^{k-1}\mathbf{m}^{k-1} \left(\mathbf{m}^{k-1}\right)^T \left(\mathbf{R}^{k-1}\right)^T \mathrm{d}\Omega = \\
= \mathbf{R}^{k-1}\, {}^{\mathbf{m}}\mathbf{H}^{k-1} \left(\mathbf{R}^{k-1}\right)^T = \mathbf{\Lambda}^{k-1},
\end{aligned}
$$

the matrix $\left(\mathbf{R}^{k-1}\right)^T$ is the matrix of the column-wise right-eigenvectors of ${}^{\mathbf{m}}\mathbf{H}^{k-1}$, and the diagonal matrix $\mathbf{\Lambda}^{k-1}$ is the matrix of the eigenvalues of ${}^{\mathbf{m}}\mathbf{H}^{k-1}$.

We finally introduce the orthogonal matrix

$$
\mathbf{Q}^{k-1} = \sqrt{\left(\mathbf{\Lambda}^{k-1}\right)^{-1}}\, \mathbf{R}^{k-1}, \tag{3}
$$

and then define the set of $\mathrm{L}^2(E)$-orthonormal polynomials that is a basis of the space $\mathbb{P}_{k-1}(E)$:

$$
\mathbf{p}^{\perp\, k-1} = \mathbf{Q}^{k-1}\mathbf{m}^{k-1}, \tag{4}
$$

with an identity mass matrix:

$$
\begin{aligned}
{}^{\mathbf{p}^{\perp}}\mathbf{H}^{k-1} = \int_E \mathbf{p}^{\perp\, k-1} \left(\mathbf{p}^{\perp\, k-1}\right)^T \mathrm{d}\Omega = \int_E \mathbf{Q}^{k-1}\mathbf{m}^{k-1} \left(\mathbf{m}^{k-1}\right)^T \left(\mathbf{Q}^{k-1}\right)^T \mathrm{d}\Omega \\
= \mathbf{Q}^{k-1}\, {}^{\mathbf{m}}\mathbf{H}^{k-1} \left(\mathbf{Q}^{k-1}\right)^T = \sqrt{\left(\mathbf{\Lambda}^{k-1}\right)^{-1}} \mathbf{\Lambda}^{k-1} \left(\sqrt{\left(\mathbf{\Lambda}^{k-1}\right)^{-1}}\right)^T = \mathbf{I}^{k-1}.
\end{aligned}
$$

Improved basis for $\mathbb{P}_k(E)$. In order to build a basis for the full space $\mathbb{P}_k(E)$ we add to the basis functions $\mathbf{p}^{\perp\, k-1}$ a set of suitable linearly independent basis functions denoted by $\mathbf{p}^{k/k-1}$, and obtained removing from the monomials $\mathbf{m}^{k/k-1}$ of order (exactly) k their components in the space of polynomials of order up $k-1$. Let us apply a Gram-Schmidt ortogonalization:

$$
\begin{aligned}
\mathbf{p}^{k/k-1} = \mathbf{m}^{k/k-1} - \left(\int_E \mathbf{m}^{k/k-1} \left(\mathbf{p}^{\perp\, k-1}\right)^T \mathrm{d}\Omega\right) \mathbf{p}^{\perp\, k-1} = \\
= \mathbf{m}^{k/k-1} - \left(\int_E \mathbf{m}^{k/k-1} \left(\mathbf{m}^{k-1}\right)^T \mathrm{d}\Omega\right) \mathbf{m}^{k-1} = \\
= \mathbf{m}^{k/k-1} - {}^{\mathbf{m}}\mathbf{H}^{k,k-1}\, \mathbf{m}^{k-1} = \left[- \left({}^{\mathbf{m}}\mathbf{H}^{k,k-1} \right)^T \quad \mathbf{I}^{k/k-1} \right] \mathbf{m}^{k}.
\end{aligned}
$$

Let us define the matrix

$$
\mathbf{R}_a^{k/k-1} = \left[- \left({}^{\mathbf{m}}\mathbf{H}^{k,k-1} \right)^T \quad \mathbf{I}^{k/k-1} \right] \in \mathbb{R}^{(n_k - n_{k-1}) \times n_{k-1}}. \tag{5}
$$

Note that the set of functions $\mathbf{p}^{k/k-1}$ is obtained starting from the set of monomials of order k, but they are general polynomials of order k orthogonal to the polynomial basis functions of order $k-1$.

Now, let us extract from these polynomials a set of linearly independent $\mathrm{L}^2(E)$ orthogonal functions ${\mathbf{p}^{\perp}}^{k/k-1}$. Let us consider the mass matrix relative to the polynomials $\mathbf{p}^{k/k-1}$

$$
{}^{\mathbf{p}^{k/k-1}}\mathbf{H}^{k/k-1} = \int_E \mathbf{p}^{k/k-1} \left(\mathbf{p}^{k/k-1}\right)^T \mathrm{d}\Omega = \\
= \mathbf{R}_a^{k/k-1} \left(\int_E \mathbf{m}^k \left(\mathbf{m}^k\right)^T \mathrm{d}\Omega \right) \left(\mathbf{R}_a^{k/k-1}\right)^T ,
$$

and let $\mathbf{R}_b^{k/k-1}$ the orthogonal matrix of change of basis that leads to a diagonal mass matrix starting from ${}^{\mathbf{p}^{k/k-1}}\mathbf{H}^{k/k-1}$:

$$
\mathbf{\Lambda}^{k/k-1} = \left(\mathbf{R}_b^{k/k-1}\right) \left({}^{\mathbf{p}^{k/k-1}}\mathbf{H}^{k/k-1} \right) \left(\mathbf{R}_b^{k/k-1}\right)^T = \\
= \left(\mathbf{R}_b^{k/k-1}\right) \left(\mathbf{R}_a^{k/k-1}\right) {}^{\mathbf{m}}\mathbf{H}^k \left(\mathbf{R}_a^{k/k-1}\right)^T \left(\mathbf{R}_b^{k/k-1}\right)^T .
$$

We, finally, define the basis functions

$$
{\mathbf{p}^{\perp}}^{k/k-1} = \sqrt{\left(\mathbf{\Lambda}^{k/k-1}\right)^{-1}} \mathbf{R}_b^{k/k-1} \mathbf{R}_a^{k/k-1} \mathbf{m}^k = \mathbf{Q}^{k/k-1} \mathbf{m}^k , \tag{6}
$$

and the new full "almost $\mathrm{L}^2(E)$-othonormal" basis is

$$
{\mathbf{p}^{\perp}}^k = \mathbf{Q}^k \mathbf{m}^k , \tag{7}
$$

where, defined the zero-matrix $\mathbf{O}^{k-1,k} \in \mathbb{R}^{n_{k-1} \times n_k - n_{k-1}}$, the matrix $\mathbf{Q}^k$ has the following structure:

$$
\mathbf{Q}^k = \begin{bmatrix} \mathbf{Q}^{k-1} & \mathbf{O}^{k-1,k} \\ & \mathbf{Q}^{k/k-1} \end{bmatrix} , \tag{8}
$$

and, in exact arithmetic, the resulting mass matrix is

$$
{}^{\mathbf{p}^{\perp}}\mathbf{H}^k = \int_E {\mathbf{p}^{\perp}}^k \left({\mathbf{p}^{\perp}}^k\right)^T \mathrm{d}\Omega = \mathbf{Q}^k \, {}^{\mathbf{m}}\mathbf{H}^k \left(\mathbf{Q}^k\right)^T = \\
= \begin{bmatrix} \mathbf{I}^{k-1} & {}^{\mathbf{p}^{\perp}}\mathbf{H}^{k-1,k} \\ {}^{\mathbf{p}^{\perp}}\mathbf{H}^{k,k-1} & \mathbf{I}^{k/k-1} \end{bmatrix} . \tag{9}
$$

For badly shaped elements, the computation of the eigenvalues-eigenvectors can be affected by a non negligible numerical error. When this happens, the diagonal blocks of the matrix ${}^{\mathbf{p}^{\perp}}\mathbf{H}^k$ are no longer identity matrices, and, for this reason, in Section 5 we consider the following definitions:

$$
{}^{\mathbf{p}^{\perp}}\mathbf{H}^{k-1} = \mathbf{Q}^{k-1} \, {}^{\mathbf{m}}\mathbf{H}^{k-1} \left(\mathbf{Q}^{k-1}\right)^T , \tag{10}
$$

$$
{}^{\mathbf{p}^{\perp}}\mathbf{H}^k = \mathbf{Q}^k \, {}^{\mathbf{m}}\mathbf{H}^k \left(\mathbf{Q}^k\right)^T , \tag{11}
$$

with the matrices $\mathbf{Q}^{k-1}$ and $\mathbf{Q}^k$ given by (3) and (8), respectively.

3.2. Computation of the projector operator matrices ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x$ *and* ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y$

In this section we describe how to obtain the $\mathrm{L}^2(E)$ projection of the gradient components of a VEM basis function following the description provided in [38, 40].

Let $\Pi^0_{k-1}\phi_{i,x}$ be the projection of the derivative with respect to the variable x of the VEM basis function ϕ_i. Let us write this projection with respect to the scaled monomial basis $\mathbf{m}$ and the basis $\mathbf{p}^\perp$ built in the previous section, respectively:

$$ {}^{\mathbf{m}}\Pi^0_{k-1}\,\phi_{i,x} = \left(\mathbf{m}^{k-1}\right)^T\ {}^{\mathbf{m}}\mathbf{\Pi}^0_x\,(:,i), \quad {}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\phi_{i,x} = \left(\mathbf{p}^{\perp\,k-1}\right)^T\ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\,(:,i), \tag{12} $$

and similarly for the derivatives with respect to the variable y.

Let us define the matrix ${}^{\mathbf{m}}\mathbf{E}_x$ of the $\mathrm{L}^2\,(E)$ scalar product of the x derivative of the VEM basis function ϕ_i with respect to the monomial basis $\mathbf{m}^{k-1}$ and the matrix ${}^{\mathbf{p}^\perp}\mathbf{E}_x$ with respect to the orthonormal basis $\mathbf{p}^{\perp\,k-1}$, respectively:

$$ {}^{\mathbf{m}}\mathbf{E}_x\,(l,i) = \int_E m_l\phi_{i,x}, \qquad {}^{\mathbf{p}^\perp}\mathbf{E}_x\,(l,i) = \int_E p^\perp_l\,\phi_{i,x}, $$

the relation between the two matrices is ${}^{\mathbf{p}^\perp}\mathbf{E}_x = \mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{E}_x$. Moreover, the $\mathrm{L}^2\,(E)$ projections ${}^{\mathbf{m}}\Pi^0_{k-1}\,\phi_{i,x}$ and ${}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\phi_{i,x}$ are defined by the systems of equations

$$ \int_E \mathbf{m}^{k-1}\left({}^{\mathbf{m}}\Pi^0_{k-1}\,\phi_{i,x}\right)\mathrm{d}\Omega = \int_E \mathbf{m}^{k-1}\phi_{i,x}\mathrm{d}\Omega, \tag{13} $$

$$ \int_E \mathbf{p}^{\perp\,k-1}\left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\phi_{i,x}\right)\mathrm{d}\Omega = \int_E \mathbf{p}^{\perp\,k-1}\phi_{i,x}\mathrm{d}\Omega, \tag{14} $$

respectively. Let us write the projections in (13), (14) by (12), we have

$$ \int_E \mathbf{m}^{k-1}\phi_{i,x}\mathrm{d}\Omega = \left(\int_E \mathbf{m}^{k-1}\left(\mathbf{m}^{k-1}\right)^T\mathrm{d}\Omega\right)\ {}^{\mathbf{m}}\mathbf{\Pi}^0_x\,(:,i), $$

$$ \int_E \mathbf{p}^{\perp\,k-1}\phi_{i,x}\mathrm{d}\Omega = \left(\int_E \mathbf{p}^\perp\left(\mathbf{p}^{\perp\,k-1}\right)^T\mathrm{d}\Omega\right)\ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\,(:,i), $$

that is

$$ {}^{\mathbf{m}}\mathbf{E}_x\,(:,i) = {}^{\mathbf{m}}\mathbf{H}^{k-1}\ {}^{\mathbf{m}}\mathbf{\Pi}^0_x\,(:,i), \qquad {}^{\mathbf{p}^\perp}\mathbf{E}_x\,(:,i) = {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\,(:,i), $$

$$ {}^{\mathbf{m}}\mathbf{E}_x = {}^{\mathbf{m}}\mathbf{H}^{k-1}\ {}^{\mathbf{m}}\mathbf{\Pi}^0_x\,, \qquad {}^{\mathbf{p}^\perp}\mathbf{E}_x = {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\,, $$

and

$$ {}^{\mathbf{m}}\mathbf{\Pi}^0_x = \left({}^{\mathbf{m}}\mathbf{H}^{k-1}\right)^{-1}\ {}^{\mathbf{m}}\mathbf{E}_x\,, \quad {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x = \left({}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\right)^{-1}\ {}^{\mathbf{p}^\perp}\mathbf{E}_x\,. \tag{15} $$

In exact arithmetic we have

$$ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x = {}^{\mathbf{p}^\perp}\mathbf{E}_x = \mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{E}_x\,, \tag{16} $$

and proceeding in a similar way we get ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y = {}^{\mathbf{p}^\perp}\mathbf{E}_y$. For the computation of the matrices ${}^{\mathbf{m}}\mathbf{E}_x$ and ${}^{\mathbf{m}}\mathbf{E}_y$ resorting to the VEM-dofs we refer to [37, 38] and

remark that, by the Green formula, all these computations can be written in term of integrals on the elements of polynomials of order $k-2$ that are VEM dofs and integrals on the boundary of VEM basis functions and polynomials of order $k-1$. In the computations performed in the following we use the expressions

$$
\begin{aligned}
{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x &= \left({}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\right)^{-1}\mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{E}_x = \mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{\Pi}^0_x\,, \qquad (17)\\
{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y &= \left({}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\right)^{-1}\mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{E}_y = \mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{\Pi}^0_y\,. \qquad (18)
\end{aligned}
$$

The matrix $\mathbf{Q}^{k-1}$ works as a preconditioner for the projection matrices ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x$ and ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y$.

3.3. Stiffness matrix computation

Denoting by Φ the column vector of the VEM basis functions ϕ_i, $i = 1, .., n_k$, let us assume that μ is a positive scalar function. The element stiffness matrix is given by

$$
\begin{aligned}
{}^{\mathbf{p}^\perp}\mathbf{K}_\mu &= \int_E \mu \left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\frac{\partial\Phi}{\partial x}\right)\left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\frac{\partial\Phi}{\partial x}\right)^T \mathrm{d}\Omega\\
&\quad + \int_E \mu \left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\frac{\partial\Phi}{\partial y}\right)\left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\frac{\partial\Phi}{\partial y}\right)^T \mathrm{d}\Omega\\
&= \int_E \mu \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\right)^T {}^{\mathbf{p}^\perp}\mathbf{p}^{k-1}\left({}^{\mathbf{p}^\perp}\mathbf{p}^{k-1}\right)^T {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\,\mathrm{d}\Omega\\
&\quad + \int_E \mu \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y\right)^T {}^{\mathbf{p}^\perp}\mathbf{p}^{k-1}\left({}^{\mathbf{p}^\perp}\mathbf{p}^{k-1}\right)^T {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y\,\mathrm{d}\Omega\\
&= \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\right)^T {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_\mu\,{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x + \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y\right)^T {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_\mu\,{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y\,,
\end{aligned}
$$

where we have defined

$$
{}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_\mu = \mu\int_E {}^{\mathbf{p}^\perp}\mathbf{p}^{k-1}\left({}^{\mathbf{p}^\perp}\mathbf{p}^{k-1}\right)^T \mathrm{d}\Omega = \mu\,\mathbf{I}^{k-1},
$$

and we can write

$$
{}^{\mathbf{p}^\perp}\mathbf{K}_\mu = \begin{bmatrix}\left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\right)^T & \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y\right)^T\end{bmatrix}\begin{bmatrix}{}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_\mu & 0\\ 0 & {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_\mu\end{bmatrix}\begin{bmatrix}{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\\ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y\end{bmatrix}. \qquad (19)
$$

In case μ is a simmetric positive definite tensor whose components are denoted by $\mu_{x_ix_j}$ with $i, j = 1, 2$ and the usual convention $x_1 = x$ and $x_2 = y$, we define

$$
\begin{aligned}
{}^{\mathbf{m}}\mathbf{H}^{k-1}_{\mu_{x_ix_j}} &= \int_E \mu_{x_ix_j}\mathbf{m}^{k-1}\left(\mathbf{m}^{k-1}\right)^T \mathrm{d}\Omega,\\
{}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\mu_{x_ix_j}} &= \int_E \mu_{x_ix_j}{}^{\mathbf{p}^\perp}\mathbf{p}^{k-1}\left({}^{\mathbf{p}^\perp}\mathbf{p}^{k-1}\right)^T \mathrm{d}\Omega = \mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{H}^{k-1}_{\mu_{x_ix_j}}\left(\mathbf{Q}^{k-1}\right)^T,
\end{aligned}
$$

and proceeding in a similar way we finally get

$$
{}^{\mathbf{p}^\perp}\mathbf{K}_\mu = \begin{bmatrix}\left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\right)^T & \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y\right)^T\end{bmatrix}\begin{bmatrix}{}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\mu_{x_1x_1}} & {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\mu_{x_1x_2}}\\ {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\mu_{x_2x_1}} & {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\mu_{x_2x_2}}\end{bmatrix}\begin{bmatrix}{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x\\ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y\end{bmatrix} \qquad (20)
$$

3.4. Computation of the projector operator $\Pi_k^\nabla : V_k(E) \to \mathcal{P}_k(E)$

First let us recall the definition of the Π_k^∇ operator [37, 40, 41]:

$$\left(\nabla\Pi_k^\nabla v_h, \nabla q_k\right) = \left(\nabla v_h, \nabla q_k\right), \qquad \forall q_k \in \mathcal{P}_k(E). \tag{21}$$

Equation (21) defines the projection $\Pi_k^\nabla v_h$ of the VEM function v_h up to a constant that can be fixed prescribing a projector operator onto constants such that $P_0 : V_k(E) \to \mathcal{P}_k(E)$:

$$P_0\Pi_k^\nabla v_h = P_0 v_h.$$

Several options for this operator are possible. As in [37, 40] we choose

$$\begin{cases} (P_0 v_h, 1)_{\partial E} = (v_h, 1)_{\partial E}, & \text{for } k = 1, \\ (P_0 v_h, 1)_E = (v_h, 1)_E, & \text{for } k \geq 2. \end{cases} \tag{22}$$

Since $\Pi_k^\nabla \phi_i \in \mathbb{P}_k(E)$ we can represent it with respect to the bases $\mathbf{m}$ and $\mathbf{p}^\perp$, with coefficients ${}^{\mathbf{m}}\mathbf{\Pi}_k^\nabla(:,i)$ and ${}^{\mathbf{p}^\perp}\mathbf{\Pi}_k^\nabla(:,i)$, respectively

$$\Pi_k^\nabla \phi_i = \left(\mathbf{m}^k\right)^T {}^{\mathbf{m}}\mathbf{\Pi}_k^\nabla(:,i) = \left(\mathbf{m}^k\right)^T \left(\mathbf{Q}^k\right)^T {}^{\mathbf{p}^\perp}\mathbf{\Pi}_k^\nabla(:,i).$$

With the monomial basis we get

$$\int_E \nabla^T \mathbf{m}^k \nabla \Pi_k^\nabla \phi_i \mathrm{d}\Omega = \int_E \nabla^T \mathbf{m}^k \nabla {\mathbf{m}^k}^T \mathrm{d}\Omega \, {}^{\mathbf{m}}\mathbf{\Pi}_k^\nabla(:,i) = {}^{\mathbf{m}}\tilde{\mathbf{G}} \, {}^{\mathbf{m}}\mathbf{\Pi}_k^\nabla(:,i) =$$
$$= \int_E \nabla^T \mathbf{m}^k \nabla \phi_i \mathrm{d}\Omega = {}^{\mathbf{m}}\tilde{\mathbf{B}}(:,i).$$

Whereas, using the basis of polynomials ${\mathbf{p}^\perp}^k$

$${}^{\mathbf{p}^\perp}\tilde{\mathbf{G}} \, {}^{\mathbf{p}^\perp}\mathbf{\Pi}_k^\nabla(:,i) = \mathbf{Q}^k \, {}^{\mathbf{m}}\tilde{\mathbf{G}} \left(\mathbf{Q}^k\right)^T {}^{\mathbf{p}^\perp}\mathbf{\Pi}_k^\nabla(:,i) = {}^{\mathbf{p}^\perp}\tilde{\mathbf{B}}(:,i) = \mathbf{Q}^k \, {}^{\mathbf{m}}\tilde{\mathbf{B}}(:,i). \tag{23}$$

The first row and first column of the matrix ${}^{\mathbf{m}}\tilde{\mathbf{G}}$ is trivially vanishing appearing in the integrals the gradient of constants. We can say that the matrix ${}^{\mathbf{p}^\perp}\tilde{\mathbf{G}}$ is singular as well. For this reason we define the matrices ${}^{\mathbf{m}}\mathbf{G}$ and ${}^{\mathbf{p}^\perp}\mathbf{G}$ in the following way. As in [40], let us consider the matrix ${}^{\mathbf{m}}\tilde{\mathbf{G}}$ and replace its first row with the vector $P_0\left(\mathbf{m}^k\right)^T$ obtaining the matrix ${}^{\mathbf{m}}\mathbf{G}$, and replace the first row of ${}^{\mathbf{m}}\tilde{\mathbf{B}}$ with $P_0\left(\Phi\right)^T$, obtaining ${}^{\mathbf{m}}\mathbf{B}$. The undetermined linear system ${}^{\mathbf{m}}\tilde{\mathbf{G}} \, {}^{\mathbf{m}}\mathbf{\Pi}_k^\nabla = {}^{\mathbf{m}}\tilde{\mathbf{B}}$ is replaced by

$${}^{\mathbf{m}}\mathbf{G} \, {}^{\mathbf{m}}\mathbf{\Pi}_k^\nabla = {}^{\mathbf{m}}\mathbf{B}. \tag{24}$$

Instead of computing ${}^{\mathbf{p}^\perp}\mathbf{G}$ by the transformation ${}^{\mathbf{p}^\perp}\mathbf{G} = \mathbf{Q}^k \, {}^{\mathbf{m}}\mathbf{G} \left(\mathbf{Q}^k\right)^T$ we could directly compute the matrix ${}^{\mathbf{p}^\perp}\mathbf{G}$ by performing a QR-rank-revealing factorization of the matrix ${}^{\mathbf{p}^\perp}\tilde{\mathbf{G}} = \mathbf{Q}^k \, {}^{\mathbf{m}}\tilde{\mathbf{G}} \left(\mathbf{Q}^k\right)^T$, and then by replacing the row of the matrix corresponding to the lowest singular value with the the vector $P_0\left({\mathbf{p}^\perp}^k\right)^T = P_0\left(\mathbf{m}^k\right)^T \left(\mathbf{Q}^k\right)^T$ and the corresponding element of the right hand side ${}^{\mathbf{p}^\perp}\tilde{\mathbf{B}} = \mathbf{Q}^k \, {}^{\mathbf{m}}\tilde{\mathbf{B}}$ with $P_0\Phi^T$, we get

$${}^{\mathbf{p}^\perp}\mathbf{G} \, {}^{\mathbf{p}^\perp}\mathbf{\Pi}_k^\nabla = {}^{\mathbf{p}^\perp}\mathbf{B}. \tag{25}$$

3.5. Computation of the projector operator matrices ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} : V_k(E) \to \mathcal{P}_k(E)$

In this section we describe how to obtain the $\mathrm{L}^2(E)$ projection of a VEM basis function following the description provided in [38, 41].

Let $\Pi^0_{k-1}\phi_i$ be the projection of the VEM basis function ϕ_i. Let us write this projection with respect to the scaled monomial basis $\mathbf{m}$ and the basis $\mathbf{p}^\perp$ built in the previous section, respectively:

$$ {}^{\mathbf{m}}\Pi^0_{k-1}\,\phi_i = \left(\mathbf{m}^{k-1}\right)^T\,{}^{\mathbf{m}}\mathbf{\Pi}^0_{k-1}\,(:,i), \quad {}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\phi_i = \left(\mathbf{p}^{\perp\,k-1}\right)^T\,{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1}\,(:,i). \tag{26} $$

Let us define the matrix ${}^{\mathbf{m}}\mathbf{C}$ of the $\mathrm{L}^2\,(E)$ scalar product of the VEM basis function ϕ_i with respect to the monomial basis $\mathbf{m}^{k-1}$ and the matrix ${}^{\mathbf{p}^\perp}\mathbf{C}$ with respect to the basis $\mathbf{p}^{\perp\,k-1}$, respectively:

$$ {}^{\mathbf{m}}\mathbf{C}\,(l,i) = \int_E m_l\phi_i, \qquad {}^{\mathbf{p}^\perp}\mathbf{C}\,(l,i) = \int_E p^\perp_l\,\phi_i, \quad l = 1,\dots n_{k-1} $$

the relation between the two matrices is ${}^{\mathbf{p}^\perp}\mathbf{C} = \mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{C}$. In the definition of the VEM space we ask that $(q,\phi_i)_E = (q,\Pi^\nabla_k\phi_i)_E$, $\forall q \in \mathbb{P}_k\,(E)\,/\mathbb{P}_{k-2}\,(E)$ and this is the way we can compute the last row of the matrix ${}^{\mathbf{m}}\mathbf{C}$ and consequantly the matrix ${}^{\mathbf{p}^\perp}\mathbf{C}$ [40, 41]. Moreover, the $\mathrm{L}^2\,(E)$ projections ${}^{\mathbf{m}}\Pi^0_{k-1}\,\phi_i$ and ${}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\phi_i$ are defined by the systems of equations

$$ \int_E \mathbf{m}^{k-1}\,{}^{\mathbf{m}}\Pi^0_{k-1}\,\phi_i \mathrm{d}\Omega = \int_E \mathbf{m}^{k-1}\phi_i \mathrm{d}\Omega, \tag{27} $$

$$ \int_E \mathbf{p}^{\perp\,k-1}\,{}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\phi_i \mathrm{d}\Omega = \int_E \mathbf{p}^{\perp\,k-1}\phi_i \mathrm{d}\Omega, \tag{28} $$

respectively. Let us write the projections in (27), (28) by (26), we have

$$ {}^{\mathbf{m}}\mathbf{C}\,(:,i) = {}^{\mathbf{m}}\mathbf{H}^{k-1}\,{}^{\mathbf{m}}\mathbf{\Pi}^0_{k-1}\,(:,i), \qquad {}^{\mathbf{p}^\perp}\mathbf{C}\,(:,i) = {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\,{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1}\,(:,i), $$

$$ {}^{\mathbf{m}}\mathbf{C} = {}^{\mathbf{m}}\mathbf{H}^{k-1}\,{}^{\mathbf{m}}\mathbf{\Pi}^0_{k-1}\,, \qquad {}^{\mathbf{p}^\perp}\mathbf{C} = {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\,{}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1}\,, $$

and

$$ {}^{\mathbf{m}}\mathbf{\Pi}^0_{k-1} = \left({}^{\mathbf{m}}\mathbf{H}^{k-1}\right)^{-1}\,{}^{\mathbf{m}}\mathbf{C}\,, $$

$$ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} = \left({}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\right)^{-1}\,{}^{\mathbf{p}^\perp}\mathbf{C} = {}^{\mathbf{p}^\perp}\mathbf{C} = \mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{C}\,. $$

From a numerical point of view, in the following, we prefer to use

$$ {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} = \left({}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\right)^{-1}\,{}^{\mathbf{p}^\perp}\mathbf{C} = \left({}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}\right)^{-1}\mathbf{Q}^{k-1}\,{}^{\mathbf{m}}\mathbf{C}\,. \tag{29} $$

3.6. Advection matrix computation

Let us consider the elemental matrix of the advection term

$$
\begin{aligned}
{}^{\mathbf{p}^\perp}\mathbf{K}_\beta \;=& \int_E \beta_x \left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\Phi \right) \left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\frac{\partial \Phi}{\partial x} \right)^T \mathrm{d}\Omega \\
&+ \int_E \beta_y \left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\Phi \right) \left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\frac{\partial \Phi}{\partial y} \right)^T \mathrm{d}\Omega = \\
&= \int_E \beta_x \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} \right)^T \mathbf{p}^{\perp\,k-1} \left(\mathbf{p}^{\perp\,k-1} \right)^T {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x \,\mathrm{d}\Omega \\
&+ \int_E \beta_y \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} \right)^T \mathbf{p}^{\perp\,k-1} \left(\mathbf{p}^{\perp\,k-1} \right)^T {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y \,\mathrm{d}\Omega = \\
&= \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} \right)^T {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\beta_x} \, {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_x + \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} \right)^T {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\beta_y} \, {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_y
\end{aligned}
$$

where, with $i = 1, 2$, we have defined

$$
\begin{aligned}
{}^{\mathbf{m}}\mathbf{H}^{k-1}_{\beta_{x_i}} &= \int_E \beta_{x_i} \mathbf{m}^{k-1} \left(\mathbf{m}^{k-1} \right)^T \mathrm{d}\Omega, \\
{}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\beta_{x_i}} &= \int_E \beta_{x_i} \mathbf{p}^{\perp\,k-1} \left(\mathbf{p}^{\perp\,k-1} \right)^T \mathrm{d}\Omega = \mathbf{Q}^{k-1}\, {}^{\mathbf{m}}\mathbf{H}^{k-1}_{\beta_{x_i}} \left(\mathbf{Q}^{k-1} \right)^T .
\end{aligned}
$$

3.7. Reaction matrix computation

Let us consider the elemental matrix of the reaction term

$$
\begin{aligned}
{}^{\mathbf{p}^\perp}\mathbf{K}_\gamma \;=& \int_E \gamma \left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\Phi \right) \left({}^{\mathbf{p}^\perp}\Pi^0_{k-1}\,\Phi \right)^T \mathrm{d}\Omega = \\
&= \int_E \gamma \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} \right)^T \mathbf{p}^{\perp\,k-1} \left(\mathbf{p}^{\perp\,k-1} \right)^T {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} \,\mathrm{d}\Omega = \\
&= \left({}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} \right)^T {}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\gamma} \, {}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1} ,
\end{aligned}
$$

where we have defined

$$
\begin{aligned}
{}^{\mathbf{m}}\mathbf{H}^{k-1}_{\gamma} &= \int_E \gamma \mathbf{m}^{k-1} \left(\mathbf{m}^{k-1} \right)^T \mathrm{d}\Omega, \\
{}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}_{\gamma} &= \int_E \gamma \mathbf{p}^{\perp\,k-1} \left(\mathbf{p}^{\perp\,k-1} \right)^T \mathrm{d}\Omega = \mathbf{Q}^{k-1}\, {}^{\mathbf{m}}\mathbf{H}^{k-1}_{\gamma} \left(\mathbf{Q}^{k-1} \right)^T .
\end{aligned}
$$

4. Validation test

Before proceding to a detailed analysis of the effects of the basis $\mathbf{p}^\perp$ in preventing instabilities on badly shaped elements, we report some numerical results for a validation of the method, in particular aimed at showing that the expected rates of convergence are effectively obtained. Let $\Omega = (0,1) \times (0,1)$ and consider the reaction-convection-diffusion problem:

$$
\begin{cases}
-\nabla \cdot (\mu \nabla u) + \beta \cdot \nabla u + \gamma u = f & \text{in } \Omega, \\
u = 0 & \text{su } \partial\Omega,
\end{cases}
$$

where $\mu(x,y) = \begin{pmatrix} 1+y^2 & 0 \\ 0 & 1+x^2 \end{pmatrix}$ is a non-constant tensor diffusivity parameter, $\beta(x,y) = (x,-y)$ is the convection velocity, $\gamma(x,y) = xy$ is the reaction parameter and f is the right-hand-side chosen such that the solution is

$$u(x,y) = -200\sqrt{\sin(1-x/\pi)}\cos(\pi x)(1-x)(1-y)xy^2\,.$$

	k = 1	k = 2	k = 3	k = 4	k=5	k=6
$\mathrm{L}^2(\Omega)$	2.08	3.14	4.29	5.25	6.60	7.53
$\mathrm{H}^1_0(\Omega)$	1.03	2.12	3.20	4.25	5.55	6.40

Table 1: Validation test: rates of convergence on triangular mesh

	k = 1	k = 2	k = 3	k = 4	k=5	k=6
$\mathrm{L}^2(\Omega)$	1.98	3.01	3.97	4.95	6.05	6.98
$\mathrm{H}^1_0(\Omega)$	1.00	1.97	2.98	3.96	5.06	6.00

Table 2: Validation test: rates of convergence on exagon mesh

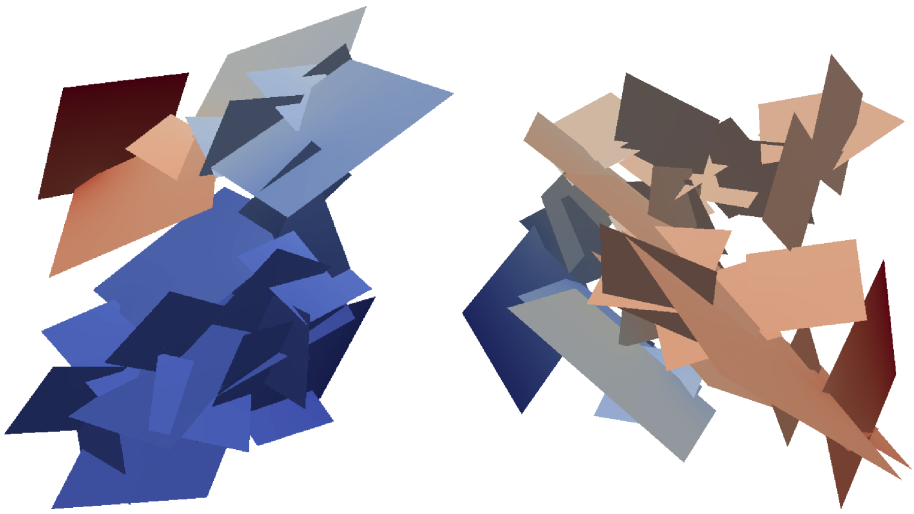
The computed rates of convergence for the norms $\mathrm{L}^2(\Omega)$ and $\mathrm{H}^1_0(\Omega)$ are reported in Tables 1 and 2 and are very close to the expected ones. Being the mesh a good quality mesh we have that the errors display the same values both with the basis $\mathbf{m}$ and $\mathbf{p}^\perp$. The rates of convergence in Table 1 are obtained on a triangular mesh with elements of area equal to 0.1, 0.01, 0.001 and 0.0001 for $k = 1,\dots,4$, and with area equal to 0.1, 0.05, 0.01, 0.005 for $k = 5,6$, while the results in Table 2 are obtained on progressively refined meshes of mildly distorted hexagons, with diameters spanning from 0.219 to 0.0266 for orders 1 up to 5, and from 0.219 to 0.071 for order 6.

5. Numerical results on Discrete Fracture Networks

In this section we consider a computational framework where instabilities arise when performing high order simulations, namely the computation of the hydraulic head inside Discrete Fracture Networks. These kind of domains are used in geomechanics to model fractured media in those cases where the rock matrix can be considered fully impervious: fractures are seen as planar polygons that intersect in the three-dimensional space, and the intersections are commonly called traces (see Figure 1 for a visualization of the DFNs that are considered in the following).

In practical applications, DFNs are generated randomly to respect the properties of the medium, which can be estimated experimentally, and are then used, for example, to determine certain quantities of interest through uncertainty quantification techniques [26].

In [2, 3, 5], the use of polygonal meshes in the VEM framework is exploited to obtain meshes which are conforming to traces, starting from an independent



(a) DFN 27 (b) DFN 36

Figure 1: The DFNs considered for numerical tests

order	minimum aspect ratio	**m** polygons	ill-conditioned polygons	badly shaped polygons	both causes
5	150	4256	124	66	9
5	50	4177	115	145	18
5	10	3775	60	547	73
6	150	3193	1187	43	32
6	50	3143	1149	93	70
6	10	2888	947	348	272

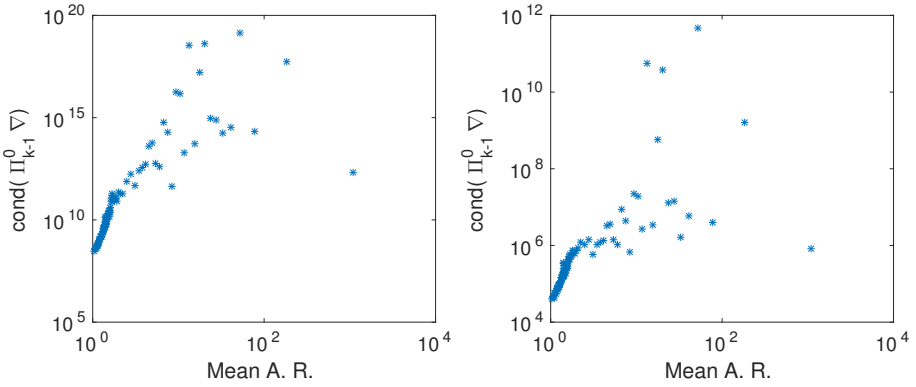
Table 3: DFN 27. Number of polygons where orthogonal polynomials were used and the motivations for their use.

triangulation whose elements are then cut along the traces. Since these cuts are in fact random, the resulting polygons are convex but are likely to be very badly shaped.

In order to circumvent the mesh generation problem an optimization approach working on totally non conforming meshes was developed [21–25, 42]. In this section we show that the use of orthogonal polynomials as described in the previous sections can prevent instabilities caused by a very large condition number of the Projector matrices arising from the use of high order VEM on badly shaped polygons.

5.1. Mesh Generation process on the DFN fractures

In this subsection we briefly recall the process described in [2]: we refer the reader to this reference for a detailed description. A starting triangular mesh is



(a) Standard basis (b) Orthogonal basis

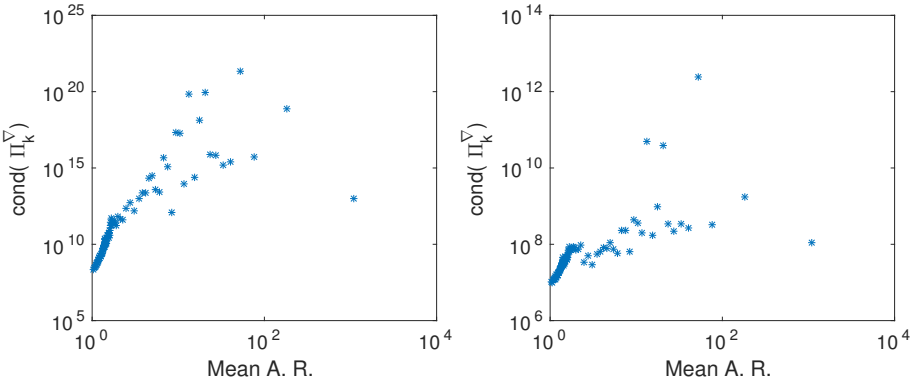
Figure 2: DFN 27, order 6. Mean condition number of the matrix representation of $\Pi^0_{k-1}\nabla$ and its standard deviation.

generated on each fracture independently of traces (fracture intersections) position. The next process of polygonal mesh generation consists of the generation of a fracture-local mesh conforming to the traces, obtained splitting the triangles of the baseline mesh into polygons conforming to the traces, iteratively for all the traces. In this step if a trace ends within an original triangle or in one of the children polygons we extend the cut segment of this trace up to the next edge. In this operation the trace is unchanged: only the segment that is cutting the polygons is extended. All the points generated by intersections between cut segments and mesh edges are added to the mesh as new vertices. At the end of this step we have a polygonal mesh on each fracture that is *locally conforming* with the traces. Finally, for each couple of intersecting fractures F_i and F_j, generating the trace T_l, we consider on the trace the union of the mesh points coming from at least one of the two fractures that are on T_l. On each fracture, polygon edges lying on T_l are accordingly split in several aligned edges at the newly added points. In such a process we, first, generate a forest of polygons with root in the original triangles. Then, we modify the leaves polygons with edges on the traces converting the edges on the traces with the aligned edges generated by the mesh points on the trace of the twin fracture.

We remark that applying a preliminary mesh smoothing step as described in [3] the aspect ratio of many elements can be strongly reduced, nevertheless in these kind of applications the geometry can unavoidably produce very badly shaped elements whatever is the conforming mesh generation and smoothing process performed. In order to consider the worst possible cases, in the presented simulations we decide not to apply any mesh smoothing step.

5.2. Problem formulation on the DFN

The computation of the hydraulic head on the DFN is provided by the solution of coupled problems on each fracture. The model we are considering



(a) Standard basis

(b) Orthogonal basis

Figure 3: DFN 27, order 6. Mean condition number of the matrix representation of Π_k^∇ and its standard deviation.

order	minimum aspect ratio	**m** polygons	ill-conditioned polygons	badly shaped polygons	both causes
4	150	4465	22	49	3
4	50	4373	15	141	10
4	10	3874	1	640	24
5	150	4322	165	38	14
5	50	4234	154	126	25
5	10	3795	80	565	99

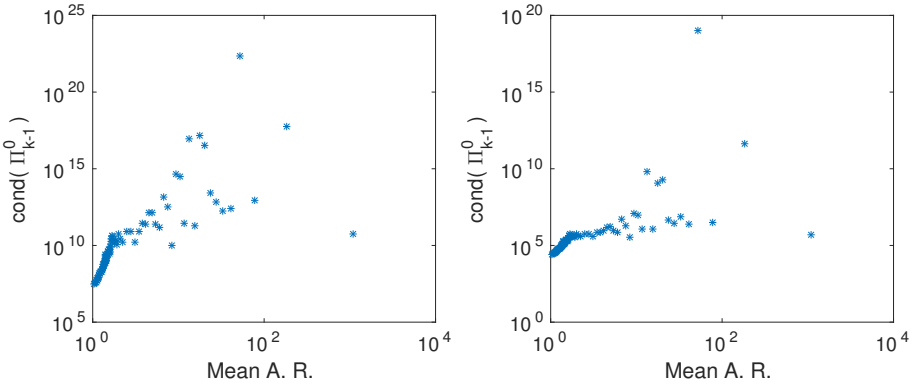
Table 4: DFN 36. Number of polygons where orthogonal polynomials were used and the motivations for their use.

is a simple Darcy model for the flow. Let $\mathcal{I}$ be the set of the indices of all the fractures in the DFN. The hydraulic head is given by the following equations $\forall i \in \mathcal{I}$:

$$\begin{cases} -\nabla \cdot (\mu \nabla h) = 0 & \text{in } F_i, \\ h = h_D & \text{on } \partial F_{i,D}, \\ \nabla h \cdot \hat{\mathbf{n}} = 0 & \text{on } \partial F_{i,N}, \end{cases}$$

where $\partial F_{i,D}$ is the subset of the boundary of the fracture F_i with Dirichlet boundary conditions and $\partial F_{i,N}$ is the subset of the boundary of the fracture F_i with Neumann boundary conditions.

Continuity matching conditions for the solution h are imposed at the traces as in [2]. We set a non-homogeneous Dirichlet boundary condition on one side of a source fracture and a homogeneous Dirichlet condition on one side of a sink fracture and homogeneous Neumann boundary conditions on all the other fracture-sides of the DFN.



(a) Standard basis (b) Orthogonal basis

Figure 4: DFN 27, order 6. Mean conditioning number of the matrix representation of Π^0_{k-1} and its standard deviation.

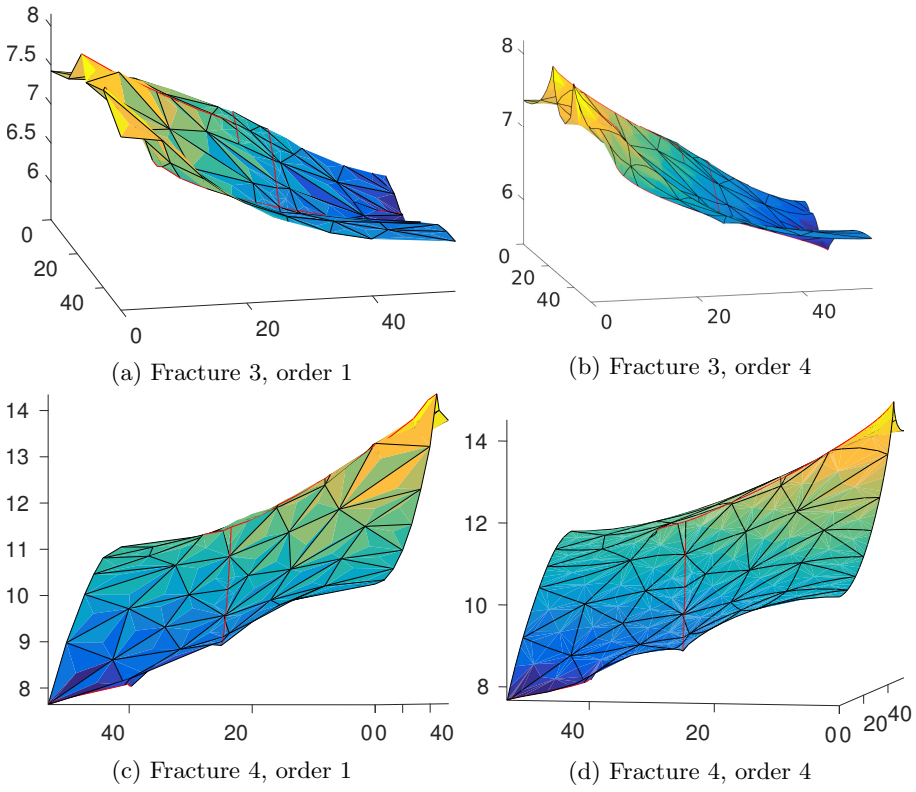
order	2	3	4	5	6
error $> 1e-4$	0	6	98	1105	4124
error > 1	0	0	8	29	352
error > 10	0	0	1	5	48
error > 100	0	0	0	1	6
max. orthog. error	$1.59 \cdot 10^{-10}$	$9.92 \cdot 10^{-01}$	$1.18 \cdot 10$	$1.77 \cdot 10^{3}$	$5.28 \cdot 10^{2}$

Table 5: DFN 36. Counts of the elements with large orthogonalization error and maximum orthogonalization error for different orders.

5.3. DFN 27

We first consider a DFN composed by 27 fractures and displaying 57 traces (see Figure 1a). Starting from a mesh of triangular elements with area smaller than 60, we have created the globally conforming VEM polygonal mesh and assembled the linear system. We first focus on the condition numbers of the several projection matrices needed for the solution of the problem.

In Figures 2-4 we report the behaviour of the condition numbers of the projectors ${}^{\mathbf{m}}\mathbf{\Pi}^0_{k-1}\nabla$, ${}^{\mathbf{m}}\mathbf{\Pi}^{\nabla}_{k}$, ${}^{\mathbf{m}}\mathbf{\Pi}^0_{k-1}$, ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1}\nabla$, ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^{\nabla}_{k}$, ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1}$, for different aspect ratios of the VEM polygonal elements using VEM of order 6. In order to draw these plots we compute the aspect ratio, defined as the ratio of the largest distance over the smallest distance between any couple of vertices of the polygon, of all the elements in the DFN and partition the full range of aspect ratios in 100 intervals uniformly. In the plots we report the mean condition numbers computed on all the elements with an aspect ratio in each of these intervals. In Figures 2 we compare the conditioning of ${}^{\mathbf{m}}\mathbf{\Pi}^0_{k-1}\nabla$ (left) and ${}^{\mathbf{p}^\perp}\mathbf{\Pi}^0_{k-1}\nabla$ (right), and we can appreciate a strong reduction of the mean condition numbers induced by the use of the basis $\mathbf{p}^\perp$. We can came to the same conclusion observing Figure 3, concerning the projector used in the VEM stabilization, as well as Figure 4.



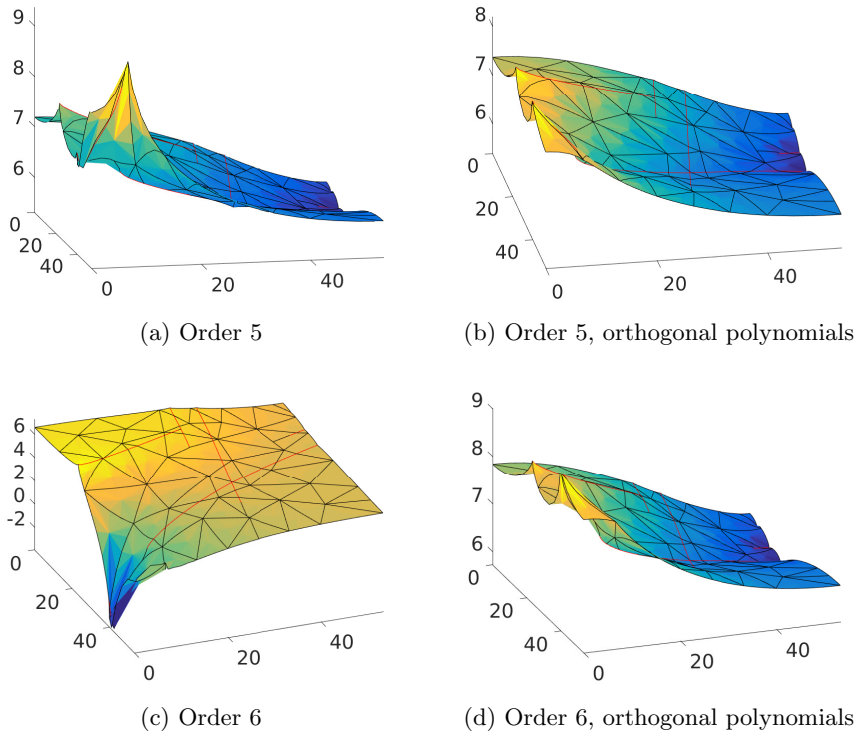
(a) Fracture 3, order 1

(b) Fracture 3, order 4

(c) Fracture 4, order 1

(d) Fracture 4, order 4

Figure 5: DFN 27. Reference solutions with low order VEM.



(a) Order 5

(b) Order 5, orthogonal polynomials

(c) Order 6

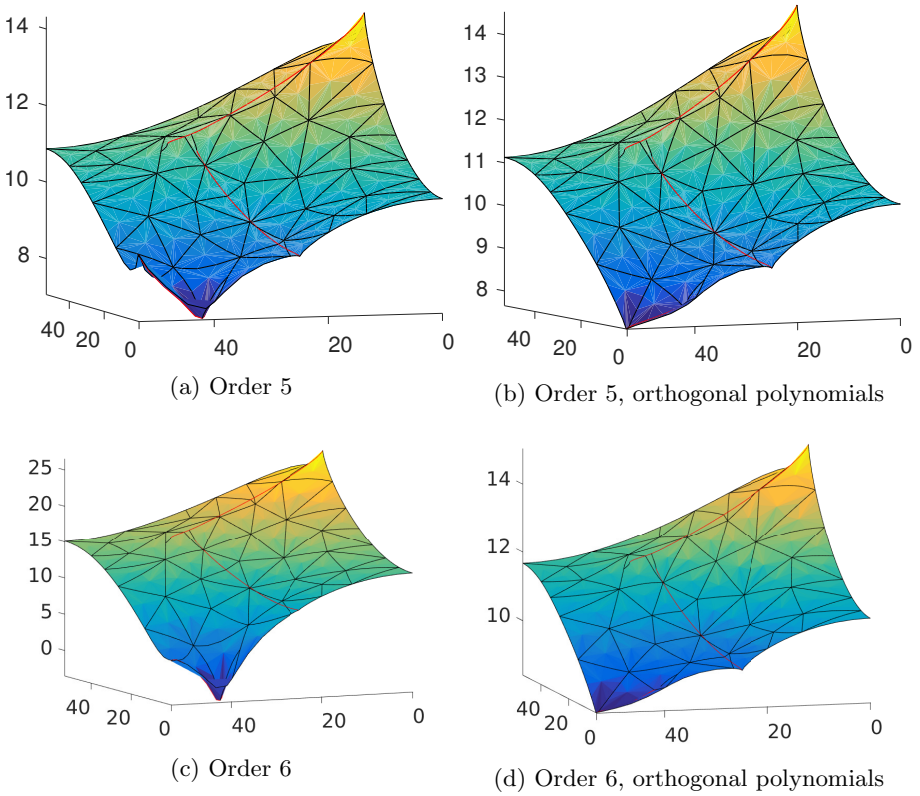
(d) Order 6, orthogonal polynomials

Figure 6: DFN 27, Fracture 3. Solutions with increasing VEM order using standard VEM and behaviour of orthogonal polynomials in correcting the instabilities

In the following figures we report some examples of the destabilizing effects of the bad conditioned projectors and the improved solution obtained with the new basis. In Figure 5 we show the low order solutions on two fractures in the DFN (Fracture 3 and Fracture 4) obtained with $k = 1$ and 4. Comparing these pictures we can appreciate the improved quality of the solution. In Figure 6 we report the solution obtained on Fracture 3 with $k = 5$ and 6. Observing Figures 6a and 6c compared with Figures 5a and 5b, we can appreciate the instabilities arising due to the ill conditioning of the local matrices with respect to the monomial basis, that gets higher as the VEM order increases. We can see that both the shape of the solution and the values are completely wrong. In Figures 6b and 6d we can see that the use of the basis $\mathbf{p}^{\perp}$ has a clear stabilizing effect. The same conclusion can be driven observing Figure 7 compared with Figures 5c and 5d.

For these results, orthogonal polynomials are used only on those polygons such that the conditioning number of the local ${}^{\mathbf{m}}\mathbf{H}^{k-1}$ is larger than 10^{10} or such that the aspect ratio is larger than 150.

In Table 3 we report the number of polygons for which orthogonal polynomials are used for different threshold values on the aspect ratio, ranging from 10



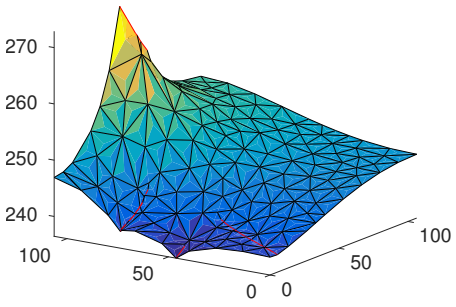
(a) Order 5

(b) Order 5, orthogonal polynomials

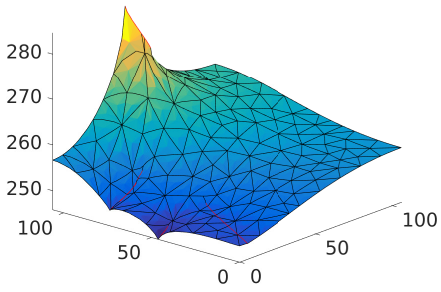
(c) Order 6

(d) Order 6, orthogonal polynomials

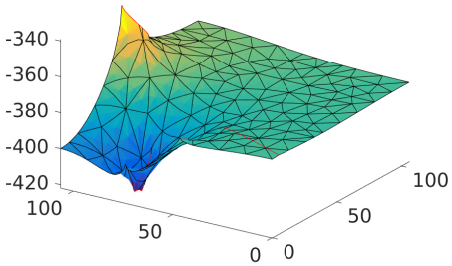
Figure 7: DFN 27, Fracture 4. Solutions with increasing VEM order using standard VEM and behaviour of orthogonal polynomials in correcting the instabilities



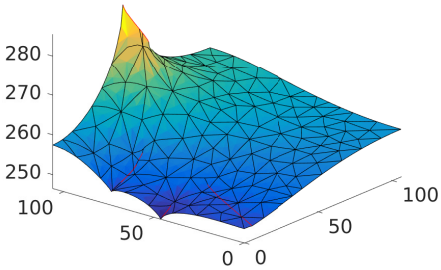
(a) Order 1



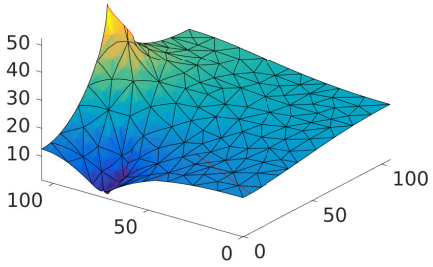
(b) Order 3



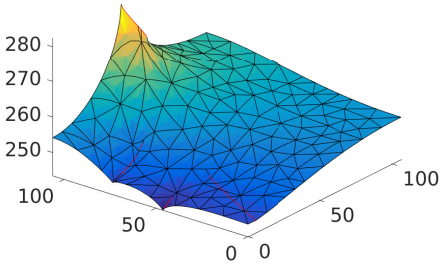
(c) Order 4



(d) Order 4, orthogonal polynomials



(e) Order 5



(f) Order 5, orthogonal polynomials

Figure 8: DFN 36, Fracture 27. Solutions with increasing VEM order using standard VEM and behaviour of orthogonal polynomials in correcting the instabilities

to 150. The third column reports the number of polygons of the mesh where the monomial basis $\mathbf{m}$ is used, the fourth column reports the number of polygons on which the basis $\mathbf{p}^{\perp}$ is introduced only due to the large conditioning of the mass matrix ${}^{\mathbf{m}}\mathbf{H}^{k-1}$, in the fifth column the number of polygons on which $\mathbf{p}^{\perp}$ is used due to the large aspect ratio of the element. In the last column we report the number of polygons that require $\mathbf{p}^{\perp}$ for both the previous reasons.

5.4. DFN 36

Our second test considers a 36 fracture network with 65 traces. We focus on two particular fractures, where instabilities arise on high order VEM and observe, in Figures 8 and 9, how the use of the proposed basis for the space of polynomials in the construction of the projectors prevents the generation of non-physical oscillations. We notice that, although using the monomial basis the shape of the solution seems correct, its values are completely wrong (see Figures 8c, 8e, 9c and 9e). Again, the figures refer to the choice of applying the change of basis only on those polygons where ${}^{\mathbf{m}}\mathbf{H}^{k-1}$ displays a condition number larger than 10^{10} or with an aspect ratio greater than or equal to 150. In Table 4 we show how the condition number of the matrix ${}^{\mathbf{m}}\mathbf{H}^{k-1}$ is influenced by the shape of the polygons and the VEM order, and the number of elements on which the change of basis is applied. We notice again that it is sufficient to apply the change of basis only locally on certain polygons to cure global instabilities.

The proposed approach is effective for this DFN up to the VEM order 5, but it fails to stabilize the solution for VEM of order 6. Indeed, in Figure 10 we see that instabilities are still present even using orthogonal polynomials on all the elements (compare Figures 10a-10b with Figures 8a-8b and Figures 10c-10d with Figures 9a-9b). This behaviour is related to the ill conditioning of some of the mass matrices ${}^{\mathbf{m}}\mathbf{H}^{k-1}$ that induces a large approximation error in the computation of the eigenvectors, that leads to a largely polluted polynomial basis. We remark that these situations can be easily detected by an evaluation of the orthogonalization error on each element:

$$\left\| \mathbf{Q}^{k-1}\, {}^{\mathbf{m}}\mathbf{H}^{k-1}\, {\mathbf{Q}^{k-1}}^{T} - \mathbf{I}^{k-1} \right\|_{\infty}. \tag{30}$$

In Figure 11 we report the orthogonalization error with respect to the aspect ratio of the elements, and in Figure 12 the orthogonalization error is plot with respect to the condition number of ${}^{\mathbf{m}}\mathbf{H}^{k-1}$. As expected, we can notice an evident correlation between them. We can remark that when these orthogonalization errors become large the generation of the orthogonal basis is not reliable and the method should be applied prudently. We can notice that for order 5 the orthogonalization error is large, but the method provides a basis for the space of polynomials that is still better than the scaled monomial basis. This is because only few elements are affected by a large error. In Table 5, we report the number of elements in the DFN with an orthogonalization error larger than $1.0E-4$, 1, 10, 100 for $k = 1, ..., 6$, and in the last row the largest orthogonalization error. In order to be more accurate also on problematic elements, in the computations

we use equation (10) for the computation of ${}^{\mathbf{p}^\perp}\mathbf{H}^{k-1}$ instead of the identity matrix in order to take advantage from all that situations in which the basis $\mathbf{p}^{\perp\,k-1}$ is no longer orthogonal, but provides a better conditioned mass matrix. As a rule of thumb we can say that when the largest orthogonalization error is not large or large orthogonalization errors occur on very few elements the method can be used, otherwise the computations cannot be considered reliable.

Finally, to further assess the behaviour of the method, we show in Figure 13 the effect of the change of basis on the conditioning of the matrices representing the projectors $\Pi^0_{k-1}\nabla$, Π^∇_k and Π^0_{k-1}, respectively. These graphs show the mean condition number with respect to the aspect ratio of the elements. We see how the use of orthogonal polynomials strongly mitigates the dependance of the condition number on the aspect ratio.

6. Conclusions

Dealing with problems with very complex geometries can easily lead to very strong mesh generation problems. In these situations the use of more flexible polygonal methods is very helpful. The VEM is a suitable and effective approach for the discretization of Partial Differential Equations. Nevertheless, in some of these applications the polygonal mesh generated for the VEM applications can suffer for very low quality elements. An applicative example in which these situations are likely to happen is in geophysical simulations following the DFN model. For the most badly shaped elements the use of the classical monomial basis for the construction of the local matrices can lead to large problems due to the large condition number of the local matrices.

In this paper, for high order VEM, we have presented the construction of a polynomial basis that leads to better conditioned local matrices and more accurate solutions. The construction is based on a local eigenvalue-eigenvector computation. This approach is very effective for very badly shaped elements, but for some elements with a huge aspect ratio the eigenvalue-eigenvector problem can be inaccurate and also this approach does not provide a reliable solution.

We have reported the success of the method in providing good solutions in some applications and have provided a criterion to evaluate the reliability of the method when the most problematic elements are met. The method has also the attractive property to be simply added to a standard VEM implementation and can be applied selectively only on the elements that really need an improvement in term of accuracy of the computations, and provides an indicator that alerts the user when the method is no longer reliable.

Acknowledgments

We wish to thank Matías Fernando Benedetto for his crucial help in developing the code that was used for numerical simulations.

References

[1] M. F. Benedetto, S. Berrone, S. Pieraccini, S. Scialò, The virtual element method for discrete fracture network simulations, Comput. Methods Appl. Mech. Engrg. 280 (0) (2014) 135 – 156. doi:10.1016/j.cma.2014.07.016.

[2] M. F. Benedetto, S. Berrone, S. Scialò, A globally conforming method for solving flow in discrete fracture networks using the virtual element method, Finite Elem. Anal. Des. 109 (2016) 23–36. doi:10.1016/j.finel.2015.10.003.

[3] M. Benedetto, S. Berrone, A. Borio, S. Pieraccini, S. Scialò, A hybrid mortar virtual element method for discrete fracture network simulations, J. Comput. Phys. 306 (2016) 148–166. doi:10.1016/j.jcp.2015.11.034.

[4] M. Benedetto, S. Berrone, A. Borio, S. Pieraccini, S. Scialò, Order preserving SUPG stabilization for the virtual element formulation of advection-diffusion problems, Computer Methods in Applied Mechanics and Engineering 311 (2016) 18 – 40. doi:10.1016/j.cma.2016.07.043.

[5] M. F. Benedetto, S. Berrone, A. Borio, The Virtual Element Method for underground flow simulations in fractured media, in: Advances in Discretization Methods, Vol. 12 of SEMA SIMAI Springer Series, Springer International Publishing, Switzerland, 2016, pp. 167–186.

[6] P. M. Adler, Fractures and Fracture Networks, Kluwer Academic, Dordrecht, 1999.

[7] M. C. Cacas, E. Ledoux, G. de Marsily, B. Tillie, A. Barbreau, E. Durand, B. Feuga, P. Peaudecerf, Modeling fracture flow with a stochastic discrete fracture network: calibration and validation: 1. the flow model, Water Resour. Res. 26 (1990) 479–489. doi:10.1029/WR026i003p00479.

[8] W. S. Dershowitz, C. Fidelibus, Derivation of equivalent pipe networks analogues for three-dimensional discrete fracture networks by the boundary element method, Water Resource Res. 35 (1999) 2685–2691. doi:10.1029/1999WR900118.

[9] J. Jaffré, J. E. Roberts, Modeling flow in porous media with fractures; discrete fracture models with matrix-fracture exchange, Numerical Analysis and Applications 5 (2) (2012) 162–167.

[10] J. Hyman, C. Gable, S. Painter, N. Makedonska, Conforming Delaunay triangulation of stochastically generated three dimensional discrete fracture networks: A feature rejection algorithm for meshing strategy, SIAM Journal on Scientific Computing 36 (2014) A1871–A1894. doi:10.1137/130942541.

[11] B. Nœtinger, N. Jarrige, A quasi steady state method for solving transient Darcy flow in complex 3D fractured networks, J. Comput. Phys. 231 (1) (2012) 23–38. doi:10.1016/j.jcp.2011.08.015.

[12] B. Nœtinger, A quasi steady state method for solving transient Darcy flow in complex 3D fractured networks accounting for matrix to fracture flow, J. Comput. Phys. 283 (2015) 205–223. doi:10.1016/j.jcp.2014.11.038.

[13] G. Pichot, J. Erhel, J. de Dreuzy, A mixed hybrid mortar method for solving flow in discrete fracture networks, Applicable Analysis 89 (2010) 1629 – 643. doi:10.1080/00036811.2010.495333.

[14] G. Pichot, J. Erhel, J. de Dreuzy, A generalized mixed hybrid mortar method for solving flow in stochastic discrete fracture networks, SIAM Journal on scientific computing 34 (2012) B86 – B105. doi:10.1137/100804383.

[15] G. Pichot, B. Poirriez, J. Erhel, J.-R. de Dreuzy, A Mortar BDD method for solving flow in stochastic discrete fracture networks, in: Domain Decomposition Methods in Science and Engineering XXI, Springer, 2014, pp. 99–112, Lecture Notes in Computational Science and Engineering.

[16] J.-R. de Dreuzy, G. Pichot, B. Poirriez, J. Erhel, Synthetic benchmark for modeling flow in 3D fractured media, Computers & Geosciences 50 (0) (2013) 59 – 71.

[17] C. Fidelibus, G. Cammarata, M. Cravero, Hydraulic characterization of fractured rocks. In: Abbie M, Bedford JS (eds) Rock mechanics: new research., Nova Science Publishers Inc., New York, 2009.

[18] C. Fidelibus, The 2D hydro-mechanically coupled response of a rock mass with fractures via a mixed bem-fem technique, International Journal for Numerical and Analytical Methods in Geomechanics 31 (11) (2007) 1329–1348.

[19] M. Cravero, C. Fidelibus, A code for scaled flow simulations on generated fracture networks, Comput. Geosci. 25 (2) (1999) 191–195.

[20] V. Lenti, C. Fidelibus, A BEM solution of steady-state flow problems in discrete fracture networks with minimization of core storage, Computers & Geosciences 29 (9) (2003) 1183 – 1190. doi:10.1016/S0098-3004(03)00140-7.

[21] S. Berrone, S. Pieraccini, S. Scialò, F. Vicini, A parallel solver for large scale DFN flow simulations, SIAM J. Sci. Comput. 37 (3) (2015) C285–C306. doi:10.1137/140984014.

[22] S. Berrone, S. Pieraccini, S. Scialò, A PDE-constrained optimization formulation for discrete fracture network flows, SIAM J. Sci. Comput. 35 (2) (2013) B487–B510. doi:10.1137/120865884.

[23] S. Berrone, S. Pieraccini, S. Scialò, On simulations of discrete fracture network flows with an optimization-based extended finite element method, SIAM J. Sci. Comput. 35 (2) (2013) A908–A935. doi:10.1137/120882883.

[24] S. Berrone, S. Pieraccini, S. Scialò, An optimization approach for large scale simulations of discrete fracture network flows, J. Comput. Phys. 256 (2014) 838–853. doi:10.1016/j.jcp.2013.09.028.

[25] S. Berrone, S. Pieraccini, S. Scialò, Towards effective flow simulations in realistic discrete fracture networks, J. Comput. Phys. 310 (2016) 181–201. doi:10.1016/j.jcp.2016.01.009.

[26] S. Berrone, C. Canuto, S. Pieraccini, S. Scialò, Uncertainty quantification in discrete fracture network models: stochastic fracture transmissivity, Comput. Math. Appl. 70 (4) (2015) 603–623. doi:10.1016/j.camwa.2015.05.013.

[27] S. Pieraccini, S. Scialò, On a PDE-constrained optimization approach for flow simulations in fractured media, in: Advances in Discretization Methods, Vol. 12 of SEMA SIMAI Springer Series, Springer International Publishing, Switzerland, 2016, pp. 27–45.

[28] S. Berrone, S. Pieraccini, S. Scialò, Non-stationary transport phenomena in networks of fractures: effective simulations and stochastic analysis, submitted.

[29] F. Brezzi, A. Buffa, K. Lipnikov, Mimetic finite differences for elliptic problems, ESAIM: Mathematical Modelling and Numerical Analysis 43 (2009) 277–295.

[30] L. Beirão da Veiga, K. Lipnikov, G. Manzini, The Mimetic Finite Difference Method for Elliptic Problems, Vol. 11 of Modeling, Simulation & Applications, Springer, 2014.

[31] F. Brezzi, L. D. Marini, Virtual element methods for plate bending problems, Computer Methods in Applied Mechanics and Engineering 253 (2013) 455 – 462. doi:10.1016/j.cma.2012.09.012.

[32] L. Beirão da Veiga, F. Brezzi, L. D. Marini, Virtual elements for linear elasticity problems, SIAM Journal on Numerical Analysis 51 (2) (2013) 794–812. doi:10.1137/120874746.

[33] L. Beirão da Veiga, C. Lovadina, D. Mora, A virtual element method for elastic and inelastic problems on polytope meshes, Computer Methods in Applied Mechanics and Engineering 295 (2015) 327 – 346. doi:10.1016/j.cma.2015.07.013.

[34] P. F. Antonietti, L. Beirão da Veiga, D. Mora, M. Verani, A stream virtual element formulation of the stokes problem on polygonal meshes, SIAM Journal on Numerical Analysis 52 (1) (2014) 386–404. doi:10.1137/13091141X.

[35] D. Mora, G. Rivera, R. Rodríguez, A virtual element method for the Steklov eigenvalue problem, Mathematical Models and Methods in Applied Sciences 25 (08) (2015) 1421–1445. doi:10.1142/S0218202515500372.

[36] L. Beirão da Veiga, C. Lovadina, A. Russo, Stability analysis for the virtual element method, arXiv preprint arXiv:1607.05988.

[37] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. Marini, A. Russo, Basic principles of virtual element methods, Mathematical Models and Methods in Applied Sciences 23 (01) (2013) 199–214.

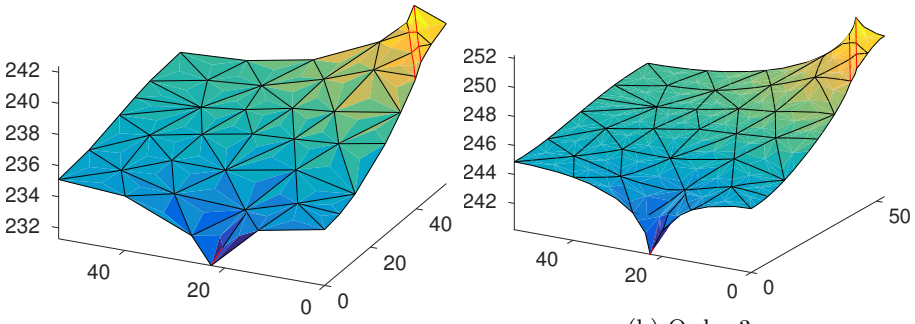
[38] L. Beirão da Veiga, F. Brezzi, L. Marini, A. Russo, Virtual element methods for general second order elliptic problems on polygonal meshes, Mathematical Models and Methods in Applied Sciences 26 (4) (2016) 729–750. doi:10.1142/S0218202516500160.

[39] A. Cangiani, G. Manzini, O. Sutton, Conforming and nonconforming virtual element methods for elliptic problems, available online at http://arxiv.org/abs/1507.03543 (2015).

[40] L. Beirão da Veiga, F. Brezzi, L. Marini, A. Russo, The hitchhiker's guide to the virtual element method, Math. Models Methods Appl. Sci 24 (8) (2014) 1541–1573.

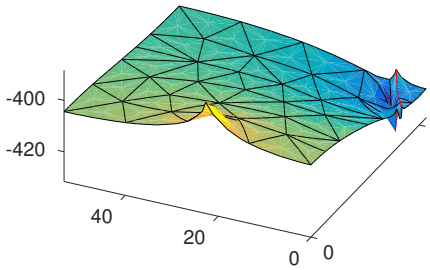
[41] B. Ahmad, A. Alsaedi, F. Brezzi, L. Marini, A. Russo, Equivalent projectors for virtual element methods, Computers & Mathematics with Applications 66 (2013) 376–391.

[42] S. Berrone, A. Borio, S. Scialò, A posteriori error estimate for a PDE-constrained optimization formulation for the flow in DFNs, SIAM J. Numer. Anal. 54 (1) (2016) 242–261. doi:10.1137/15M1014760.

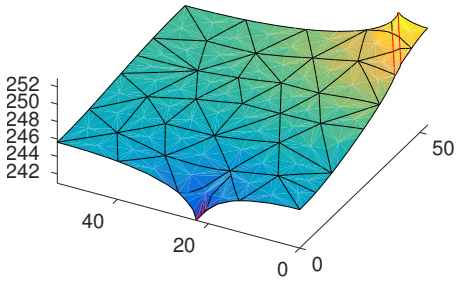


(a) Order 1

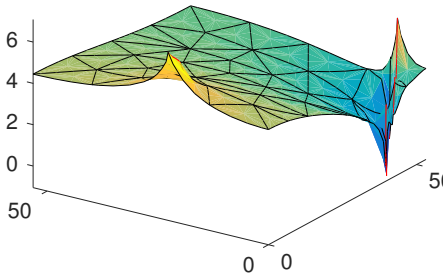
(b) Order 3



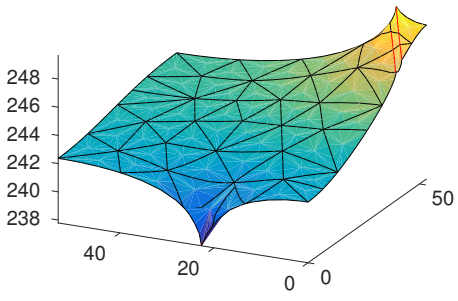
(c) Order 4



(d) Order 4, orthogonal polynomials

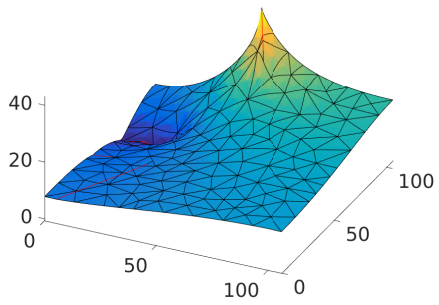


(e) Order 5

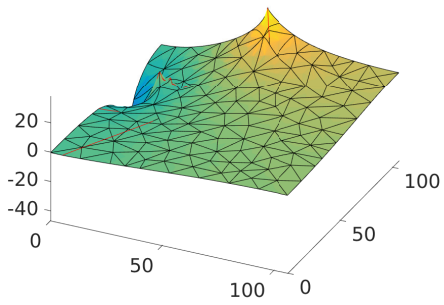


(f) Order 5, orthogonal polynomials

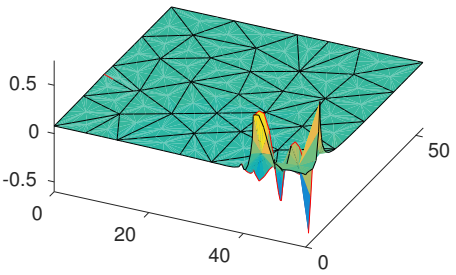
Figure 9: DFN 36, Fracture 29. Solutions with increasing VEM order using standard VEM and behaviour of orthogonal polynomials in correcting the instabilities



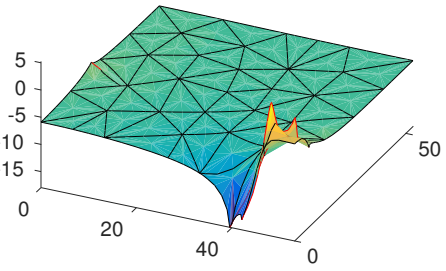
(a) Fracture 27, standard basis



(b) Fracture 27, orthogonal polynomials

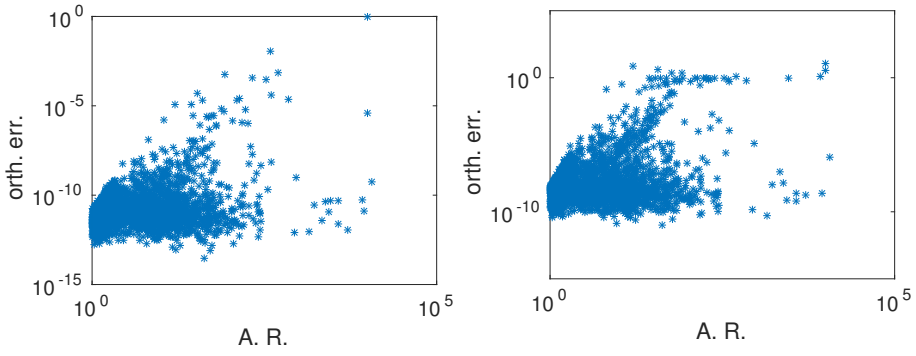


(c) Fracture 29, standard basis



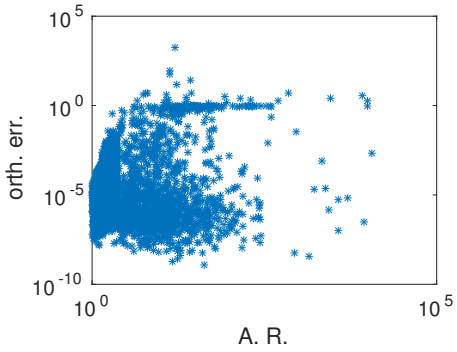
(d) Fracture 29, orthogonal polynomials

Figure 10: DFN 36, order 6. Solutions using standard VEM polynomial basis and orthogonal polynomials

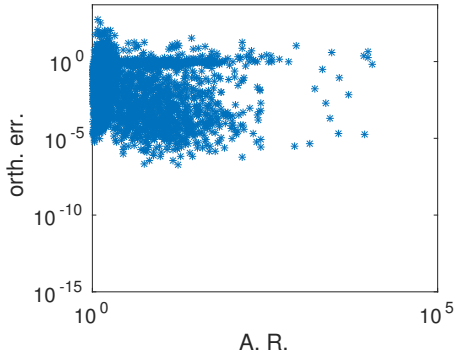


(a) Order 3

(b) Order 4

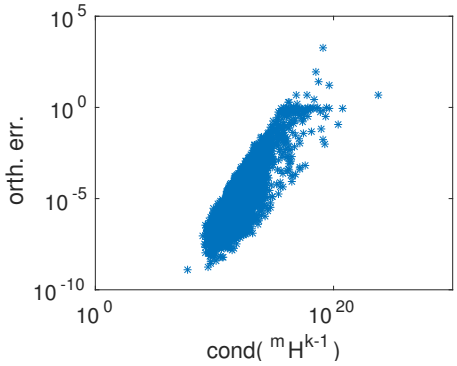


(c) Order 5

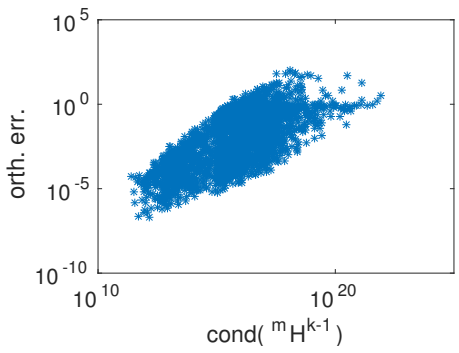


(d) Order 6

Figure 11: DFN 36: error of orthogonalization of ${}^{\mathbf{m}}\mathbf{H}^{k-1}$ vs. aspect ratio

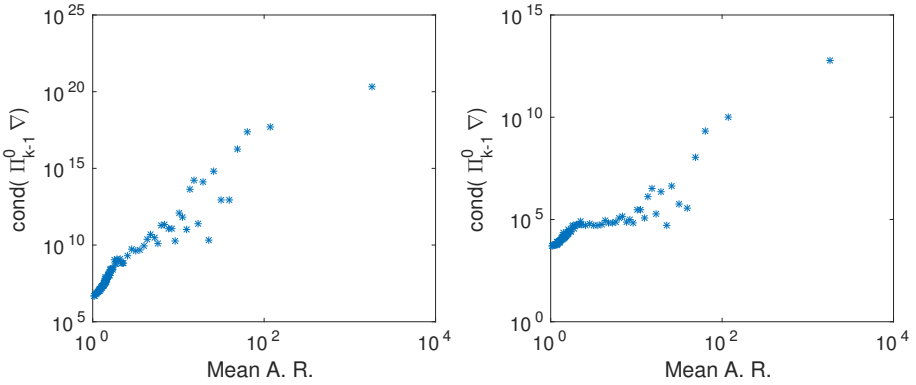


(a) Order 5



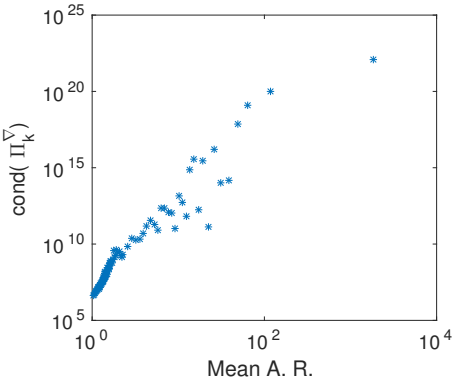
(b) Order 6

Figure 12: DFN 36: error of orthogonalization of ${}^{\mathbf{m}}\mathbf{H}^{k-1}$ vs. its condition number

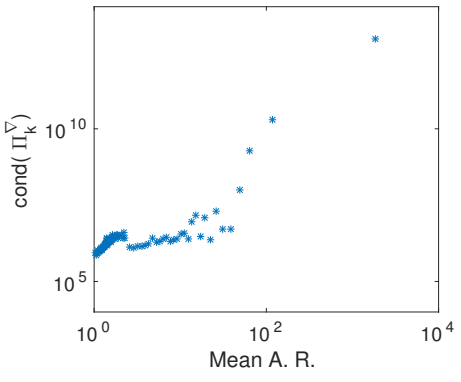


(a) Standard basis, $k = 5$

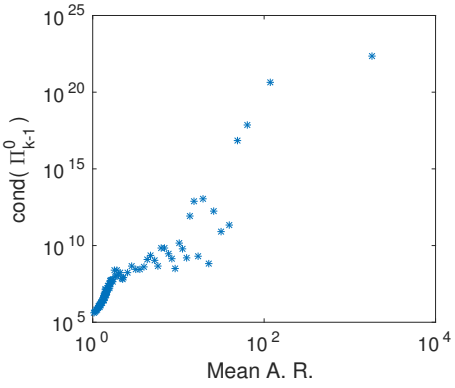
(b) Orthogonal basis, $k = 5$



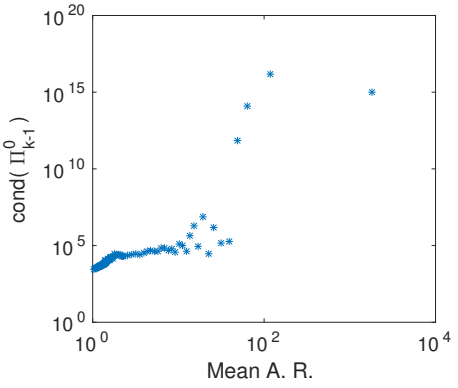
(c) Standard basis, $k = 5$



(d) Orthogonal basis, $k = 5$



(e) Standard basis, $k = 5$



(f) Orthogonal basis, $k = 5$

Figure 13: DFN 36, order 5. Mean condition number and standard deviation of $\Pi^0_{k-1}\nabla$, $\Pi^0_{k-1}\nabla$ and Π^0_{k-1}.