

A globally conforming method for solving flow in discrete fracture networks using the Virtual Element Method

Original

A globally conforming method for solving flow in discrete fracture networks using the Virtual Element Method / Benedetto, MATIAS FERNANDO; Berrone, Stefano; Scialo', Stefano. - In: FINITE ELEMENTS IN ANALYSIS AND DESIGN. - ISSN 0168-874X. - STAMPA. - 109:(2016), pp. 23-36. [10.1016/j.finel.2015.10.003]

Availability:

This version is available at: 11583/2602373 since: 2016-03-18T13:02:42Z

Publisher:

Elsevier

Published

DOI:10.1016/j.finel.2015.10.003

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

A globally conforming method for solving flow in discrete fracture networks using the Virtual Element Method

Matías Fernando Benedetto^a, Stefano Berrone^{a,*}, Stefano Scialò^a

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

Abstract

A new approach for numerically solving flow in Discrete Fracture Networks (DFN) is developed in this work by means of the Virtual Element Method (VEM). Taking advantage of the features of the VEM, we obtain global conformity of all fracture meshes while preserving a fracture-independent meshing process. This new approach is based on a generalization of globally conforming Finite Elements for polygonal meshes that avoids complications arising from the meshing process. The approach is robust enough to treat many DFNs with a large number of fractures with arbitrary positions and orientations, as shown by the simulations. Higher order Virtual Element spaces are also included in the implementation with the corresponding convergence results and accuracy aspects.

Keywords: VEM, Fracture flows, Darcy flows, Discrete Fracture Networks

1. Introduction

The present work deals with a new approach based on the Virtual Element Method (VEM) for the simulation of the flow in Discrete Fracture Networks (DFNs). DFN models are one of the possible approaches for simulating subsurface flows and they consist of a set of planar polygons in 3D space resembling

*Corresponding author

Email addresses: matias.benedetto@polito.it (Matías Fernando Benedetto), stefano.berrone@polito.it (Stefano Berrone), stefano.scialo@polito.it (Stefano Scialò)

6 the fractures in the underground. Each fracture is modelled individually, as
7 opposed to continuum models with equivalent porosity, and, for geological for-
8 mations with a sparse fracture network that mainly affects the flow path, this
9 approach is recommended [1, 2]. DFNs are used in a wide range of applications
10 such as pollutant percolation, gas recovery, aquifers, reservoir analysis, among
11 others [3] [4].

12 Stationary flow in a DFN is modelled using Darcy’s law and introducing
13 a transmissivity tensor for each fracture that depends on its aperture and its
14 resistance to flow. The surrounding rock matrix is considered impervious. The
15 goal is to obtain the hydraulic head distribution in the system, which is the sum
16 of the pressure head and the elevation. Fluid can only flow through fractures
17 and across intersections between fractures, also called traces, but no tangential
18 flow is considered along traces. The hydraulic head is a continuous function,
19 but with discontinuous derivatives across the traces, which act as sources/sinks
20 of flow. More complex models for the flow in the fractures can be found in the
21 literature [5]. Since little is known about the subsurface fractures, stochastic
22 models are used in order to determine distributions of aperture, hydrological
23 properties, size, orientation, density, and aspect ratio of the fractures.

24 Geometrical complexity is the greatest challenge when dealing with DFN-
25 based simulations. Since the fracture generation has a random component, many
26 complex situations arise that render the meshing process very complicated and
27 sometimes impossible, e.g. very small angles, very close and almost parallel
28 traces, high disparity of traces lengths, etc. In order to use traditional finite
29 elements, fracture grids have to match in all the intersections between fractures,
30 since these are discontinuity interfaces for the first order derivatives of the solu-
31 tion. All the aforementioned geometrical configurations complicate the meshing
32 process and are the biggest obstacle in the discretization of the problem because
33 it becomes very computationally demanding to obtain a good mesh from such a
34 badly predisposed geometry. Furthermore, the meshing procedure depends on
35 the whole DFN and is not independent for each fracture. When a large DFN is

36 considered that can have thousands of fractures, mesh conformity requirements
 37 can lead to a very high number of elements that are far more than those de-
 38 manded by the required level of accuracy. In [6], a BEM (Boundary Element
 39 Method) was applied that aims to minimize core memory usage by defining and
 40 storing only a relation between nodal fluxes and hydraulic head on traces for
 41 each fracture. The problem of obtaining a good globally conforming mesh is
 42 the subject of ongoing research. In [7], an adaptive mesh refinement method is
 43 described that aims for a high resolving mesh. Previous works [8, 9] suggest a
 44 simplification of the geometry to ease meshing. Monodimensional pipes joining
 45 fractures, instead of traces, have been put forward as an alternative in [10] and
 46 [11]. In [12], a mixed formulation and a mesh modifying procedure was used
 47 to solve DFNs and reducing the number of elements for each fracture. Another
 48 mixed formulation was used in [13], where local corrections of traces are applied
 49 in order to obtain a globally conforming mesh. The mortar method was used to
 50 impose conditions between fractures with non-matching grids to obtain a mixed
 51 hybrid formulation in [14], with a subsequent generalization in [15] that includes
 52 trace intersections within a fracture. A novel approach was proposed in [16],
 53 [17], [18] and [19] in which the problem was reformulated as a PDE-constrained
 54 optimization. The minimization of a properly defined functional is adopted to
 55 enforce hydraulic head continuity and flux conservation at fracture intersections.
 56 Traditional finite elements (FEM) as well as extended finite elements (XFEM)
 57 were implemented to solve the problem.

58 In this work, we aim to provide an easy, natural way for generating conform-
 59 ing meshes for complex DFN problems using the VEM. The proposed approach
 60 is a generalization of traditional conforming finite elements, keeping the method
 61 as simple and streamlined as possible. Some of the ideas presented here were
 62 present in a previous work by the authors [20], that introduced Virtual Ele-
 63 ments (VEM) to DFNs. In [20] the VEM is used on locally conforming meshes
 64 and an optimization approach is adopted to handle the non-conformity of the
 65 global mesh. Here both local and global conformity is enforced, and classical

approaches, borrowed from the domain decomposition methods, can be used to solve the problem. We make absolutely no assumptions on the meshing procedure, which is done independently for each fracture and without any consideration of the position of the traces. Traces are not modified in any way, and using some of the features of the VEM, local and global conformity for the mesh is obtained by means of splitting the original elements of the meshes independently generated on each fracture into polygons of an arbitrary number of vertices.

Using Lagrange multipliers we obtain a hybrid system that can be solved with different methods, including FETI algorithms for domain decomposition.

Section 2 provides the formulation of the DFN problem in the present context, whereas a brief summary of the VEM is reported in Section 3, and in Section 4 the proposed method is described in detail. Numerical results are presented in Section 5, where some convergence results are given and the applicability of the method to DFNs is discussed.

2. The continuous problem

Let us consider a set of open convex planar polygonal fractures $F_i \subset \mathbb{R}^3$ with $i = 1, \dots, N$, with boundary ∂F . Our DFN is $\Omega = \bigcup_i F_i$, with boundary $\partial\Omega$. Even though the fractures are planar, their orientations in space are arbitrary, such that Ω is a 3D set. The set $\Gamma_D \subset \partial\Omega$ is where Dirichlet boundary conditions are imposed, and we assume $\Gamma_D \neq \emptyset$, whereas $\Gamma_N = \partial\Omega \setminus \Gamma_D$, is the portion of the boundary with Neumann boundary conditions. Dirichlet and Neumann boundary conditions are prescribed by the functions $h^D \in H^{\frac{1}{2}}(\Gamma_D)$ and $g^N \in H^{-\frac{1}{2}}(\Gamma_N)$ on the Dirichlet and Neumann part of the boundary, respectively. We further set $\Gamma_{iD} = \Gamma_D \cap \partial F_i$, $\Gamma_{iN} = \Gamma_N \cap \partial F_i$, and $h_i^D = h^D|_{\Gamma_{iD}}$ and $g_i^N = g^N|_{\Gamma_{iN}}$. The set \mathcal{T} collects all the traces, i.e. the intersections between fractures, and each trace $T \in \mathcal{T}$ is given by the intersection of exactly two fractures, $T = \bar{F}_i \cap \bar{F}_j$, such that there is a one to one relationship between a trace T and a couple of fracture indexes $\{i, j\} = \mathcal{I}(T)$. We will also denote by

95 \mathcal{T}_i the set of traces belonging to fracture F_i .

96 Subsurface flow is governed by the gradient of the hydraulic head $H =$
 97 $\mathcal{P} + \zeta$, where $\mathcal{P} = p/(\varrho g)$ is the pressure head, p is the fluid pressure, g is the
 98 gravitational acceleration constant, ϱ is the fluid density and ζ is the elevation.

99 We define the following functional spaces:

$$V_i = H_0^1(F_i) = \left\{ v \in H^1(F_i) : v|_{\Gamma_{iD}} = 0 \right\},$$

100

$$V_i^D = H_D^1(F_i) = \left\{ v \in H^1(F_i) : v|_{\Gamma_{iD}} = h_i^D \right\},$$

101 and

$$V = \left\{ v : v|_{F_i} \in V_i, \forall i = 1, \dots, N, \gamma_T(v|_{F_i}) = \gamma_T(v|_{F_j}), \forall T \in \mathcal{T}_i, \{i, j\} = \mathcal{I}(T) \right\},$$

where γ_T is the trace operator onto T . It is then possible to formulate the DFN problem, given by the Darcy's law in its weak form on the fractures with additional constraints of continuity of the hydraulic head across the traces: for $i = 1, \dots, N$, find $H_i \in V_i^D$ such that $\forall v \in V$

$$\begin{aligned} \sum_{i=1}^N \int_{F_i} \mathcal{K}_i \nabla H_i \nabla v|_{F_i} dF_i &= \sum_{i=1}^N \left(\int_{F_i} f_i v|_{F_i} dF_i + \langle g_i^N, v|_{\Gamma_{N_i}} \rangle_{H^{-\frac{1}{2}}(\Gamma_{N_i}), H^{\frac{1}{2}}(\Gamma_{N_i})} \right), \\ \gamma_T(H_i) &= \gamma_T(H_j), \forall T \in \mathcal{T}, \{i, j\} = \mathcal{I}(T) \end{aligned}$$

102 where \mathcal{K}_i is the fracture transmissivity tensor, that we assume is constant on
 103 each fracture. The second equation represents the continuity of the hydraulic
 104 head across traces. On each fracture of the DFN the following bilinear form
 105 $a_i : V_i \times V_i \mapsto \mathbb{R}$ is defined as:

$$a_i(H_i, v|_{F_i}) = \int_{F_i} \mathcal{K}_i \nabla H_i \nabla v|_{F_i} dF_i. \quad (2.1)$$

106 3. The Virtual Element Method

107 This section provides a quick overview of the VEM, recalling the main fea-
 108 tures useful in the present context. We refer the reader to the original paper [21]
 109 for a proper introduction and to [22] for a guide on implementation. Further
 110 developments can be found in [23], [24], [25] and [26]. The VEM has also been
 111 applied to problems in elasticity [27], plate bending [28], the Stokes problem
 112 [29] and has sparked interest in other applications as well.

113 Borrowing ideas from the Mimetic Finite Difference method [30, 31], the
 114 VEM can be regarded as a generalization of regular finite elements to meshes
 115 made up by polygonal elements of any number of edges. The discrete functional
 116 space on each element has, in general, not only polynomial functions but also
 117 other functions that are only known at a certain set of degrees of freedom.
 118 Given a bilinear form to be approximated with the VEM, our goal is to build
 119 a discrete bilinear form that coincides with the exact one when at least one of
 120 the arguments is a polynomial. For the other cases, a rough approximation that
 121 scales in a desired way is enough to obtain the desired convergence qualities of
 122 the method.

123 Given a domain $F \subset \mathbb{R}^2$, a mesh τ_h on F , made of polygons $\{E\}$ with mesh
 124 parameter h (i.e. the square root of the maximum element area), and the space
 125 of the polynomials of maximum order k , \mathcal{P}_k , let us define the local space $V_{k,h}^E$
 126 for a given polynomial degree k as:

$$V_{k,h}^E = \{v_h \in H^1(E) : v_h|_{\partial E} \in C^0(\partial E), v_h|_e \in \mathcal{P}_k(e) \forall e \subset \partial E, \Delta v_h \in \mathcal{P}_{k-2}(E)\}$$

127 where ∂E is the border of E , and e an edge.

128 From the above definition it is clear that the space $\mathcal{P}_k(E)$ is a subset of $V_{k,h}^E$.

129 We define the following degrees of freedom for each element E :

- 130 • The value of v_h at the vertices of E ;
- 131 • The value of v_h at $k - 1$ internal points on each edge of E ;

132 • The moments $\frac{1}{|E|} \int_E v_h m_\alpha$ for $|\alpha| \leq k - 2$,

133 where m_α , with $\alpha = (\alpha_1, \alpha_2)$, represent scaled monomials of the type

$$m_\alpha = \left(\frac{x - x_c}{h_E}\right)^{\alpha_1} \left(\frac{y - y_c}{h_E}\right)^{\alpha_2},$$

134 and (x_c, y_c) and h_E are the centroid and the diameter of the element E respec-
 135 tively. Different choices for the second type of degree of freedom is possible
 136 instead of point values, e.g. edge moments. We have chosen point values on
 137 Gauss-Lobatto nodes on edges for numerical integration purposes. The selected
 138 set of degrees of freedom is unisolvent [21], and therefore, given an element E
 139 with n_v vertices, we have that the dimension of $V_{k,h}^E$ is $\#V_{k,h}^E = n_v k + \frac{k(k-1)}{2}$.
 140 We finally choose a basis for $V_{k,h}^E$, made of functions ϕ_i with $i = 1, \dots, \#V_{k,h}^E$,
 141 such that, calling $\text{dof}_j(v)$, for $j = 1, \dots, \#V_{k,h}^E$ the j -th degree of freedom ap-
 142 plied to v , we have $\text{dof}_j(\phi_i) = \delta_{ij}$, being δ_{ij} the Kronecker delta. The global
 143 virtual element space is:

$$V_{k,h} = \{v_h \in H^1(F) : v_h|_E \in V_{k,h}^E \ \forall E \in \tau_h\},$$

144 and we can easily check that the chosen degrees of freedom on the edges of
 145 each element allow to easily enforce continuity of any function $v_h \in V_{k,h}$ on the
 146 internal edges of the partition τ_h .

147 Let us now consider the restriction of the bilinear form (2.1) to a mesh ele-
 148 ment E , $a_i^E(.,.)$. We aim at building a discrete bilinear form $a_{i,h}^E : V_{k,h}^E \times V_{k,h}^E \mapsto$
 149 \mathbb{R} having the previously stated polynomial consistency, i.e. the discrete bilinear
 150 form has to coincide with the exact one when at least one of the arguments is
 151 a polynomial of maximum degree k . To this end let us consider the projector
 152 operator of order k on E :

$$\Pi_{E,k}^\nabla : V_{k,h}^E \longrightarrow \mathcal{P}_k(E)$$

153 such that

$$\Pi_{E,k}^\nabla q_k = q_k \text{ for all } q_k \in \mathcal{P}_k(E),$$

154 defined by the equations

$$\begin{aligned} \int_E \nabla q_k \cdot \nabla v_h &= \int_E \nabla q_k \cdot \nabla \Pi_{E,k}^\nabla v_h \text{ for all } q_k \in \mathcal{P}_k(E), \\ \int_E \Pi_{E,k}^\nabla v_h &= \int_E v_h. \end{aligned}$$

155 The projection $\Pi_{E,k}^\nabla v_h$ can be uniquely defined starting from the degrees of
 156 freedom of v_h using integration by parts [22] and represents an orthogonality
 157 condition in the H^1 inner product. The first equation defines the projection up
 158 to a constant, which is defined by the second equation. Other options for the
 159 second equation exist [26]. For order $k = 1$, it can be taken as

$$\frac{1}{N^v} \sum_{i=1}^{N^v} \Pi_{E,k}^\nabla v_h(\mathcal{V}_i) = \frac{1}{N^v} \sum_{i=1}^{N^v} v_h(\mathcal{V}_i)$$

160 where \mathcal{V}_i are the vertices of the element and N^v its number.

161 **Remark 1.** *In the case of a more complex equation than the Laplacian (or*
 162 *even the Laplacian with non-constant coefficients), other projectors have to be*
 163 *considered [26].*

164 Let us now take any symmetric, positive definite bilinear form $S_{i,h}^E : V_{k,h}^E \times$
 165 $V_{k,h}^E \mapsto \mathbb{R}$, such that there exist c_0 and c_1 positive constants, independent of the
 166 element E and its diameter, that verify

$$c_0 a^E(v_h, v_h) \leq S_{i,h}^E(v_h, v_h) \leq c_1 a^E(v_h, v_h) \quad \forall v_h \in V_{k,h}^E \text{ with } \Pi_{E,k}^\nabla v_h = 0.$$

This implies that $S_{i,h}^E$ scales like $a_i^E(v_h, v_h)$, and then the local discrete bilinear
 form $a_{i,h}^E$ is set as

$$\begin{aligned} a_{i,h}^E(u_h, v_h) &= a_i^E(\Pi_{E,k}^\nabla u_h, \Pi_{E,k}^\nabla v_h) + \\ &S_{i,h}^E(u_h - \Pi_{E,k}^\nabla u_h, v_h - \Pi_{E,k}^\nabla v_h) \quad \forall u_h, v_h \in V_{k,h}^E. \end{aligned}$$

167 The first terms ensures the *consistency* and the second one the *stability* of the
 168 form. Finally, the complete discrete bilinear form becomes

$$a_{i,h}(u_h, v_h) = \sum_{E \in \tau_h} a_{i,h}^E(u_h, v_h) \quad \forall u_h, v_h \in V_{k,h}.$$

169 A possible choice for the bilinear form $S_{i,h}^E$ is the usual Euclidean product in
 170 $\mathbb{R}^{\#V_{k,h}^E \times \#V_{k,h}^E}$ between two vectors whose components are the values of the func-
 171 tions at the degrees of freedom. A stiffness matrix K_i is associated to the discrete
 172 bilinear form $a_{i,h}$, defined as :

$$(K_i)_{pq} = a_{i,h}(\phi_q, \phi_p), \text{ for } p, q = 1, \dots, \#V_{k,h}.$$

173 In general it is not true that the VEM stiffness matrix approximates the exact
 174 stiffness matrix as if it were computed numerically.

175 For the right hand side with load term f , it is enough for optimal convergence
 176 [22] to consider

$$\begin{aligned} (f, v_h) &= \sum_{E \in \tau_h} \int_E f \Pi_{E,k-1}^0 v_h & \text{for order } k = 1, 2, \\ (f, v_h) &= \sum_{E \in \tau_h} \int_E f \Pi_{E,k-2}^0 v_h & \text{for order } k \geq 3, \end{aligned}$$

177 where $\Pi_{E,k}^0$ is the the full L^2 projection on the polynomials of degree k .

178 4. Problem implementation

179 4.1. Mesh generation

180 Mesh generation is done independently for each fracture regardless of traces
 181 and their positions. The process of mesh generation consists of three steps:
 182 the first task is the generation of a baseline triangulation of each fracture, not
 183 necessarily conforming to trace disposition, and independent on each fracture;
 184 the second step is the generation of a fracture-local conforming mesh, splitting

the triangles of the baseline mesh into polygons conforming to the traces; finally on each fracture F_i , nodes are added on the traces $T \in \mathcal{T}_i$ corresponding to the nodes of the intersecting fracture F_j with $\{i, j\} = \mathcal{I}(T)$, $\forall T \in \mathcal{T}_i$, thus gaining global conformity. The three steps are depicted in Figure 4.1, and, the second and third steps are further described in full details in the next paragraphs.

4.1.1. Local conformity

Local conformity is obtained as in the previous work [20]. Every time a trace intersects an edge of the triangulation, a new node is created there. Nodes are also created at trace tips. If a trace tip is inside a triangular element, we extend the geometrical segment coinciding with the trace up to the nearest edge of the triangulation, thereby creating a new edge and a new node. The trace is not modified, being now a subset of the extended segment. By doing this, we split the original elements of the triangulation into new convex “sub-elements”, which are elements of the mesh in their own right. The end result is a mesh of polygonal elements for which all traces are covered by element edges, see Figures 4.1a and 4.1b, where element colouring indicates the number of edges. A careful inspection of those subfigures reveals all of the situations described above.

Remark 2. *An optional mesh modification has been implemented that rearranges some of the nodes of the baseline triangulation before the splitting process, so as to make them coincide with nearby traces, trace tips and trace intersections. This leads to better shaped elements and fewer DOFs for the final mesh and it is not computationally demanding.*

4.1.2. Global conformity

After obtaining the locally conforming mesh the subsequent step is to ensure that all the nodes on the traces are included in the meshes of both fractures that share the trace. These nodes are the ones shared by more than one fracture. This is the most important feature of the method we are proposing and takes full advantage of VEM versatility. Given a trace T shared by fractures F_i and F_j , we define $U_T^{F_i}$ as the set of all nodes on the trace T in fracture F_i and analogously $U_T^{F_j}$ for F_j . The procedure used to obtain the global conforming

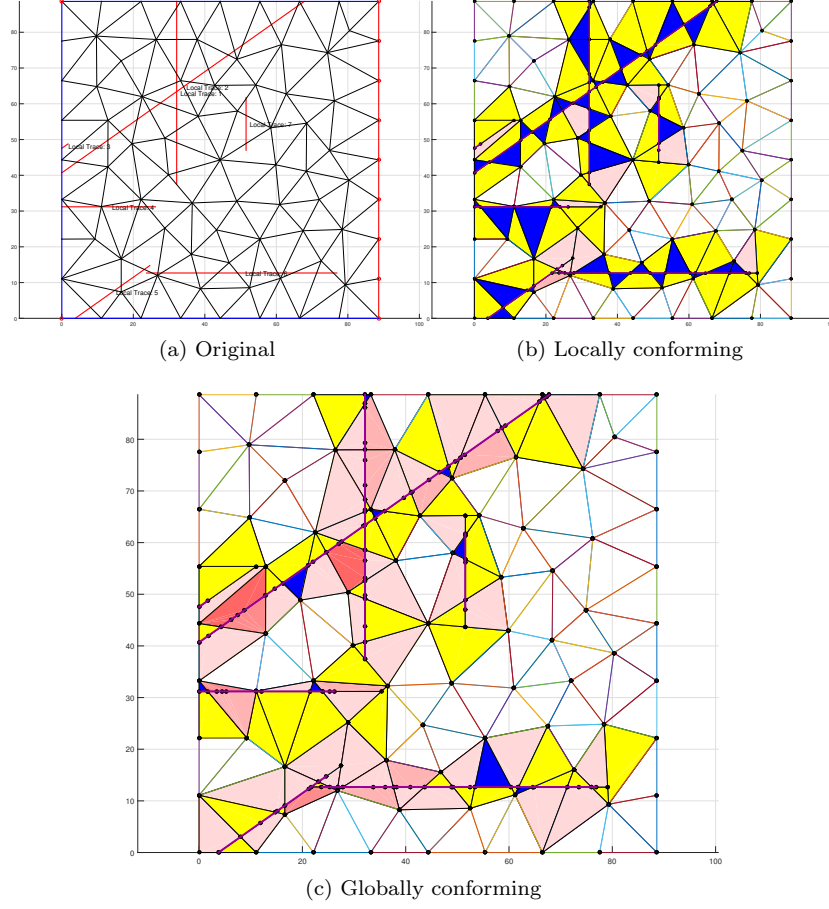


Figure 4.1: Original mesh, VEM mesh and final globally conforming mesh

216 mesh guarantees that both trace tips are included and that the discretization
 217 includes all nodes on the traces and covers it precisely. The complete trace
 218 discretization is then $U_T = U_T^{F_i} \cup U_T^{F_j}$. What remains now is to simply add
 219 the set of nodes $U_T \setminus U_T^{F_i}$ on the corresponding elements of fracture F_i and
 220 analogously for fracture F_j . This can be done since the VEM allows for elements
 221 of arbitrary number of edges and 180° angles between them. The final globally
 222 conforming mesh is shown in Figure 4.1c and is identical to the previous mesh
 223 except for the new added nodes on the traces and a change in element colouring
 224 that is an indication of the increment in the number of edges and DOFs.

225 *4.2. Imposing matching conditions*

226 For every fracture F_i , with $i = 1, \dots, N$, we call n_{dof_i} the number of DOFs
 227 of fracture F_i and we assemble the stiffness matrix $K_i \in \mathbb{R}^{n_{dof_i} \times n_{dof_i}}$ following
 228 the procedure described in Section 3. Then we construct the column vectors
 229 $f_i \in \mathbb{R}^{n_{dof_i}}$ as the vector of load values (including terms arising from non-
 230 homogeneous boundary conditions) and h_i as the vector of nodal values of the
 231 discrete solution. We note that the matrix K_i is singular for fractures with pure
 232 Neumann boundary conditions. For the complete DFN we have:

$$233 \quad K = \begin{pmatrix} K_1 & 0 & \cdots & 0 \\ 0 & K_2 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & K_N \end{pmatrix}, \quad f = \begin{pmatrix} f_1 \\ \vdots \\ \vdots \\ f_N \end{pmatrix} \quad \text{and} \quad h = \begin{pmatrix} h_1 \\ \vdots \\ \vdots \\ h_N \end{pmatrix}.$$

234 In order to obtain the saddle point linear system for the complete DFN we
 235 have to impose matching conditions for the nodes on the traces that guarantee
 236 the continuity condition of the hydraulic head. We do that by means of Lagrange
 237 multipliers λ_t , for $t = 1, \dots, n_{dof_t}$. They are introduced for each node on the
 238 traces in a non-redundant way (see [32]) which means that in the case of two
 239 intersecting traces, i.e. three fractures sharing a single point in space (as in
 240 the example of Section 5.1.2), only two multipliers are added. To each index
 241 $t = 1, \dots, n_{dof_t}$ corresponds a node on a trace T that is shared by fractures F_i
 242 and F_j , and we denote by $dof_i(t)$ the corresponding global DOF for node t on
 243 F_i and analogously by $dof_j(t)$ the DOF on F_j . We define $N^h = \sum_{i=1}^N n_{dof_i}$,
 244 and the row vector $L_t \in \mathbb{R}^{N^h}$ as:

245

$$L_t = \begin{pmatrix} & & & dof_i & & & & dof_j & & & \\ 0 & \cdots & 0 & 1 & 0 & \cdots & 0 & -1 & 0 & \cdots & 0 \end{pmatrix}. \quad (4.1)$$

246 Finally, we set $L \in \mathbb{R}^{n_{dof_t} \times N^h}$ as the matrix:

$$L = \begin{pmatrix} L_1 \\ \vdots \\ \vdots \\ L_{n_{dof_t}} \end{pmatrix}.$$

248 The final linear system is:

$$\begin{bmatrix} K & L^T \\ L & 0 \end{bmatrix} \begin{bmatrix} h \\ \lambda \end{bmatrix} = \begin{bmatrix} f \\ 0 \end{bmatrix}. \quad (4.2)$$

249 This saddle point problem has a unique solution as it can be easily proven
250 resorting to classical results of quadratic programming [33].

251 When the dimensions of the system 4.2 are large, the use of an iterative
252 method and of a preconditioner is advised. We briefly recall the one-level FETI
253 method for domain decomposition as described in [34] here implemented. In this
254 method the primal variables are determined in terms of the Lagrange multipliers.
255 More precisely, we define a block diagonal matrix R as

$$R = \begin{pmatrix} R_1 & 0 & \cdots & 0 \\ 0 & R_2 & \cdots & \vdots \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & \cdots & R_N \end{pmatrix}$$

256
257 where each sub-matrix R_i , for $i = 1, \dots, N$ is such that its columns form a
258 basis of the kernel of K_i , $\ker(K_i)$, so that $\ker(K) = \text{range}(R)$. In the case of
259 the Laplacian operator, R_i corresponds to constant solutions for the subdomains
260 with pure Neumann boundary conditions. Subdomains with Dirichlet boundary
261 conditions have a unique solution and therefore have no contribution for R . It
262 can be shown that

$$h = K^*(f - L^T \lambda) + R\alpha$$

263 where K^* is the pseudoinverse of K and the vector α depends on λ but not
 264 on the primal variables h . This means that if we solve a system for λ , this
 265 completely determines the solution. In order to solve this system for λ , a choice
 266 of several preconditioners is possible.

267 We give a brief outline of the procedure to obtain the Dirichlet preconditioner
 268 for the one-level FETI, denoted M^{-1} . Let us define \mathcal{K}^t as the sum of
 269 transmissivity values of the fractures that share the node associated with the
 270 degree of freedom t . We first multiply the coefficient $(L)_{t,dof_i(t)}$ by $\mathcal{K}_i/\mathcal{K}^t$ and
 271 the coefficient $(L)_{t,dof_j(t)}$ by $\mathcal{K}_j/\mathcal{K}^t$. This takes into account the relative weight
 272 of the transmissivity coefficient of each fracture with respect to the sum of the
 273 transmissivity coefficients of the fractures associated with that node. We collect
 274 then the new coefficients in a matrix L_D . Then, for each fracture we denote by
 275 τ the set of fracture DOFs corresponding to nodes placed on the traces, and by
 276 ζ the set of the remaining DOFs and we can rearrange matrices K_i to obtain:

$$\tilde{K}_i = \begin{bmatrix} K_i^{(\zeta\zeta)} & K_i^{(\tau\zeta)T} \\ K_i^{(\tau\zeta)} & K_i^{(\tau\tau)} \end{bmatrix}.$$

277 The local Schur complement S_i is defined as:

$$S_i = K_i^{(\tau\tau)} - K_i^{(\tau\zeta)}(K_i^{(\zeta\zeta)})^{-1}K_i^{(\tau\zeta)T}.$$

278 If we call S the block diagonal Schur complement matrix of the whole system,
 279 the Dirichlet preconditioner for the one-level FETI is:

$$M^{-1} = L_D S L_D^T.$$

280 This is called Dirichlet preconditioner as a consequence of the fact that for each
 281 application of the preconditioner a local Dirichlet problem has to be solved. The

lumped preconditioner is defined similarly as:

$$M^{-1} = L_D K^{(\tau\tau)} L_D^T,$$

where $K^{(\tau\tau)}$ is the block diagonal matrix made up by the local $K_i^{(\tau\tau)}$. We note that in order to define inner products for the Preconditioned Conjugate Gradient (PCG) FETI algorithm, a symmetric, positive definite matrix Q is used [34]. In our experiments we have considered $Q = M^{-1}$.

5. Numerical results

In this section we present some numerical results, beginning with convergence results for benchmark problems and VEM spaces of various orders. We also compare the results obtained with this approach to the results of a validated XFEM based method on a medium size DFN [18, 19]. We conclude showing some examples of numerical instabilities arising mainly with the higher order VEM approximation spaces for certain particularly adverse geometrical configurations. All of the results were obtained using a constant transmissivity tensor $\mathcal{K} = 1$ for all fractures.

5.1. Convergence results

The error norms used for the convergence curves are the usual L^2 and H^1 norms. The error is computed by taking the projection of the discrete solution on the space of polynomials, since the values of the discrete solution are only known at the DOFs and are not explicitly known inside the elements (see [26]):

$$\begin{aligned} Err_{L^2}^2 &= \sum_{E \in \mathcal{T}_\delta} \|H - \Pi_{E,k}^\nabla h_E\|_{L^2(E)}^2, \\ Err_{H^1}^2 &= \sum_{E \in \mathcal{T}_\delta} \|H - \Pi_{E,k}^\nabla h_E\|_{H^1(E)}^2 \end{aligned}$$

where $\Pi_{E,k}^\nabla$ is the projection operator of order k as defined in Section 3, H is the exact solution and h_E is the discrete solution restricted to element E .

303 The flux incoming in a fracture through the traces is computed as the jump of
 304 the conormal derivative of the discrete solution across the traces. For every trace
 305 we fix a tangential orientation and a normal unit vector obtained by clockwise
 306 rotating by 90° the tangent vector of the trace in the fracture plane. For every
 307 mesh edge $e \subset T$, i.e. an edge included in trace T , we consider a unique normal
 308 vector $\mathbf{n}_{e,i}$ in F_i with an orientation given by the normal vector fixed for the
 309 trace, and we define the flux incoming in the fracture F_i through the edge e ,
 310 named $u_{e,i}$, as follows:

$$\begin{aligned}
 u_{\text{left},e,i} &= \nabla \Pi_{E_l,k}^\nabla h_{E,i} \cdot \mathbf{n}_{e,i}, \\
 u_{\text{right},e,i} &= -\nabla \Pi_{E_r,k}^\nabla h_{E,i} \cdot \mathbf{n}_{e,i}, \\
 u_{e,i} &= u_{\text{left},e,i} + u_{\text{right},e,i},
 \end{aligned}$$

311 where E_l and E_r are the elements to the left and to the right of the trace that
 312 share the edge e , respectively.

313 The flux entering in the fracture F_i through trace T is then obtained by
 314 repeating this procedure over all the mesh edges in F_i belonging to T :

$$u_{T,i} = \sum_{e \subset T} u_{e,i}.$$

315 The L^2 error of the flux on the trace is then:

$$Err U_{L^2}^2 = \|U_{T,i} - u_{T,i}\|_{L^2(T)}^2,$$

316 where $U_{T,i}$ is the exact incoming flux in F_i through trace T .

317 5.1.1. Benchmark problem 1

318 This first problem has been considered before in the context of the XFEM
 319 (eXtended finite elements) [17] and of the VEM [20] as a single-fracture problem.
 320 Nevertheless, it remains interesting for the fact that it includes a trace tip
 321 inside the domain and the exact solution is known. In this work the problem is

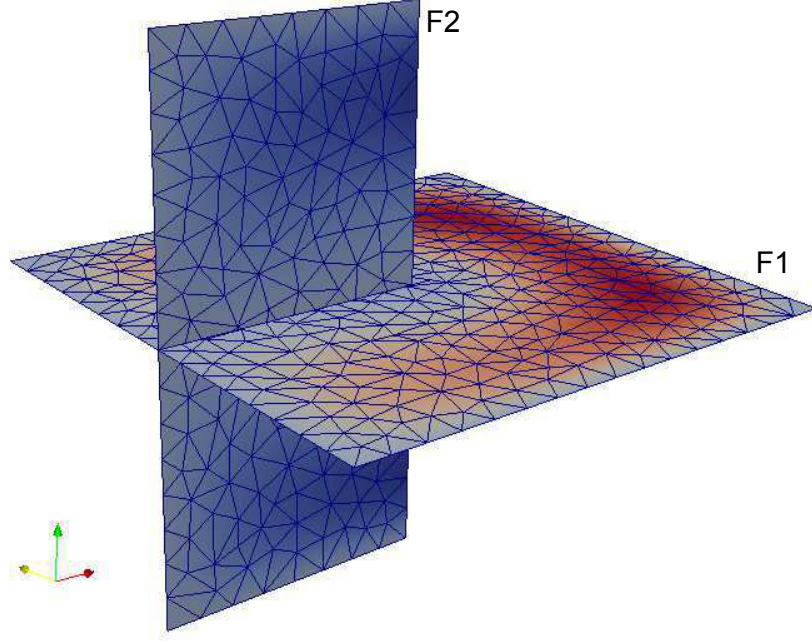


Figure 5.1: Spatial distribution of fractures for benchmark problem 1

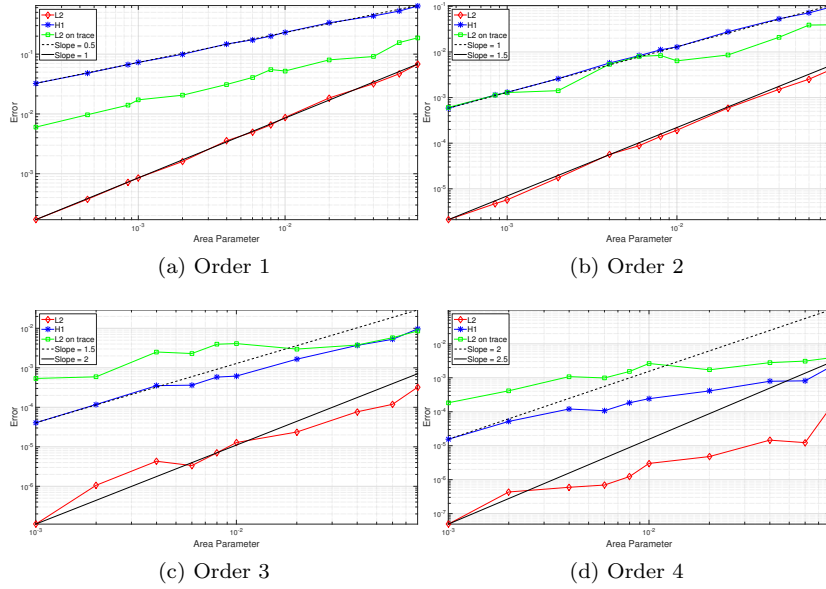


Figure 5.2: Convergence curves for benchmark problem 1 - Fracture 1

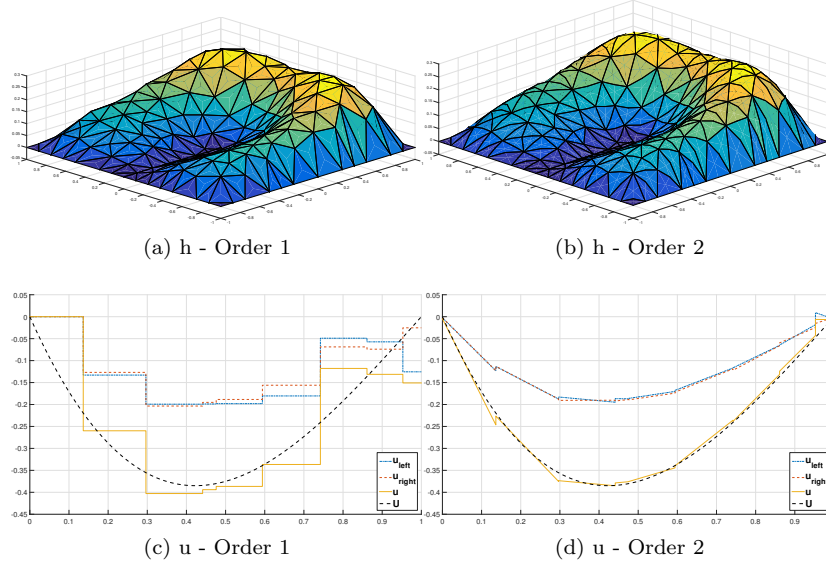


Figure 5.3: Solutions for benchmark problem 1 - Fracture 1

322 considered as a 2-fracture DFN, as shown in Figure 5.1 and the error calculations

323 and convergence curves are shown for the first fracture, F_1 .

324 Let us define the domains F_1 and F_2 as

$$F_1 = \{(x, y, z) \in \mathbb{R}^3 : -1 \leq x \leq 1, -1 \leq y \leq 1, z = 0\},$$

$$F_2 = \{(x, y, z) \in \mathbb{R}^3 : -1 \leq x \leq 0, -1 \leq z \leq 1, y = 0\},$$

325 with a single trace $T = \{(x, y) \in \mathbb{R}^3 : y = 0, z = 0 \text{ and } -1 \leq x \leq 0\}$ ending in

326 the interior of F_1 (Figure 5.1).

327 Exact solutions for F_1 and F_2 are given by $H_1^{ex}(x, y)$ and $H_2^{ex}(x, y)$:

$$H_1^{ex}(x, y, z) = -\cos\left(\frac{1}{2} \arctan2(x, y)\right) (x^2 - 1)(y^2 - 1)(x^2 + y^2)$$

$$H_2^{ex}(x, y, z) = -\cos\left(\frac{1}{2} \arctan2(x, y)\right) (z^2 - 1)(x^2 - 1)(z^2 + x^2)$$

328 where $\arctan2(x, y)$ is the arc-tangent function with 2 arguments, that re-

329 turns the appropriate quadrant of the computed angle.

The problem is then:

$$-\Delta H = -\Delta H_1^{ex} \text{ on } F_1 \setminus T,$$

$$H = 0 \text{ on } \partial F_1,$$

$$-\Delta H = -\Delta H_2^{ex} \text{ on } F_2 \setminus T,$$

$$H = (z^2 - z^4) \cos(\pi/4) \text{ on } \partial F_2^D$$

$$H = 0 \text{ on } \partial F_2 \setminus \partial F_2^D.$$

where $\partial F_2^D = \{(x, y, z) \in \mathbb{R}^3 : x = 0, y = 0, -1 \leq z \leq 1\}$ is the boundary of F_2 with non-homogeneous Dirichlet boundary conditions.

Convergence curves for the VEM of orders from 1 to 4 are shown in Figure 5.2. The expected rates of convergence are obtained for orders 1 and 2, whereas a slower rate of convergence for orders 3 and 4 was obtained as a consequence of the insufficient regularity of the exact solution in the sense of Sobolev spaces.

Numerical solutions for the hydraulic head H_1 with the VEM of orders 1 and 2 are shown in Figure 5.3 a) and b). In Figure 5.3 c) and d), we present a comparison between the exact solution and the approximate solution of the flux incoming in F_1 , as well as its left and right components. Note how the approximation of the trace flux U is piecewise constant for order 1 VEM and piecewise linear for order 2 VEM, and the approximation of the exact flux (dashed line) with the VEM of second order is greatly improved.

5.1.2. Benchmark problem 2

This problem shows the performance of the proposed approach in presence of trace intersections. The considered system consists of 3 fractures and 3 traces

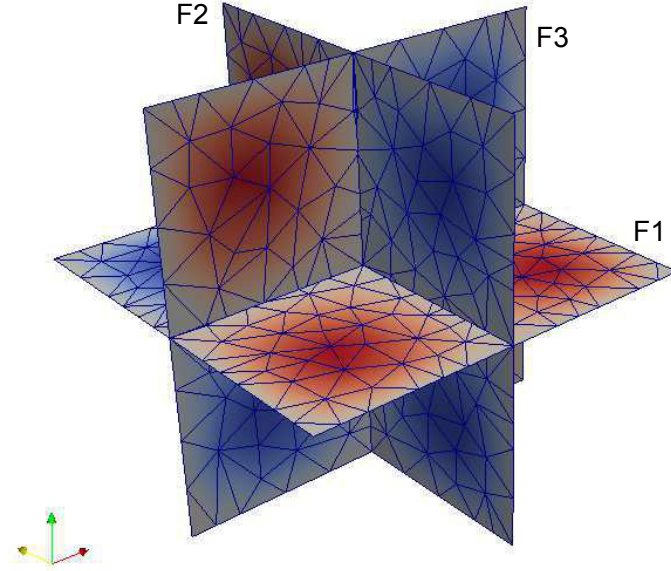


Figure 5.4: Spatial distribution of fractures for benchmark problem 2

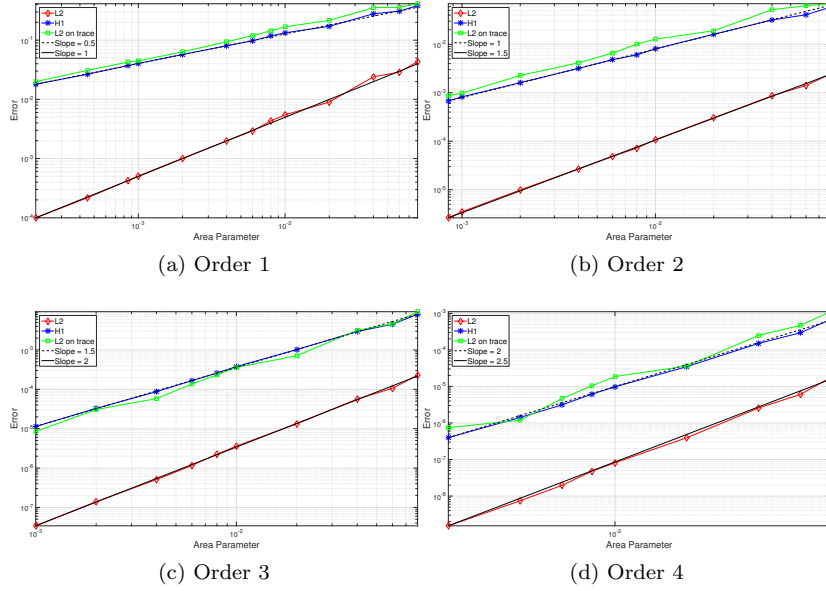


Figure 5.5: Convergence curves for benchmark problem 2 - Fracture 1

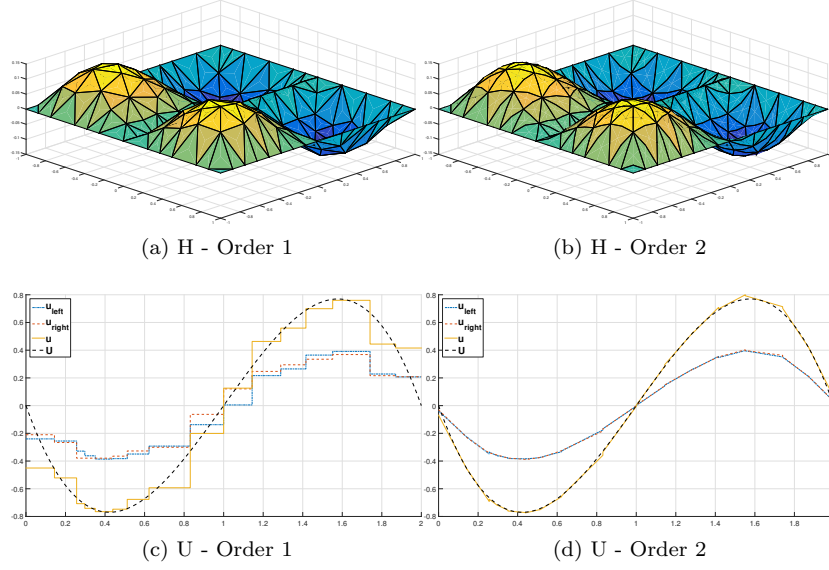


Figure 5.6: Solutions for benchmark problem 2 - Fracture 1 and trace 1

as shown in Figure 5.4:

$$F_1 = \{(x, y, z) \in \mathbb{R}^3 : -1 \leq x \leq 1, -1 \leq y \leq 1, z = 0\},$$

$$F_2 = \{(x, y, z) \in \mathbb{R}^3 : -1 \leq y \leq 1, -1 \leq z \leq 1, x = 0\},$$

$$F_3 = \{(x, y, z) \in \mathbb{R}^3 : -1 \leq z \leq 1, -1 \leq x \leq 1, y = 0\},$$

$$T_1 = \{(x, y, z) \in \mathbb{R}^3 : -1 \leq x \leq 1, y = 0, z = 0\},$$

$$T_2 = \{(x, y, z) \in \mathbb{R}^3 : -1 \leq y \leq 1, z = 0, x = 0\},$$

$$T_3 = \{(x, y, z) \in \mathbb{R}^3 : -1 \leq z \leq 1, x = 0, y = 0\}.$$

Note that all of the three traces intersect in a single point $P = (0, 0, 0)$ in space (as it is always the case for the intersection of 3 planar fractures).

Exact solutions are known for all fractures:

$$\begin{aligned}
H_1^{ex}(x, y) &= |x|(1+x)(1-x)y(1+y)(1-y), \\
H_2^{ex}(y, z) &= y(1+y)(1-y)|z|(1+z)(1-z), \\
H_3^{ex}(z, x) &= z(1+z)(1-z)x(1+x)(1-x).
\end{aligned}$$

353 Note that H_1^{ex} and H_2^{ex} are not C^1 in the whole fracture, but, for each of
 354 the 4 subdomains defined by the traces in each fracture, they are polynomials
 355 of degree 6.

The problem is then:

$$\begin{aligned}
-\Delta H &= 6|x|y(x^2 + y^2 - 2) \text{ on } F_1 \setminus \mathcal{T}_1, \\
-\Delta H &= 6|y|z(y^2 + z^2 - 2) \text{ on } F_2 \setminus \mathcal{T}_2, \\
-\Delta H &= 6zx(z^2 + y^2 - 2) \text{ on } F_3 \setminus \mathcal{T}_3, \\
H &= 0 \text{ on } \partial F_1 \cup \partial F_2 \cup \partial F_3.
\end{aligned}$$

356 Convergence curves for the VEM of orders from 1 to 4 are shown in Figure
 357 5.5 and solutions for order 1 and 2 are reported in Figure 5.6. In contrast with
 358 benchmark problem 1, the expected convergence speed is achieved for all orders,
 359 since now the exact solution has C^∞ regularity on each of the subdomains
 360 defined by the traces and the mesh for the numerical solution is conforming to
 361 the traces. This is a sufficient condition for optimal convergence rates, [35, 36].
 362 The error in the discrete solution for VEM of order 6 is $\|H - h\|_{L^2}^2 = 3.53e - 19$,
 363 $\|\partial_x(H - h)\|_{L^2}^2 = 5.09e - 18$ and $\|\partial_y(H - h)\|_{L^2}^2 = 5.85e - 18$, being then of the
 364 same order of the round-off error in double precision. This confirms that the
 365 discrete solution coincides numerically with the exact solution.

366 5.2. DFN - 27 fractures

367 Let us consider the DFN shown in Figure 5.7 consisting of 27 fractures.
 368 A sink fracture F_1 and a source fracture F_2 are defined, both having a non

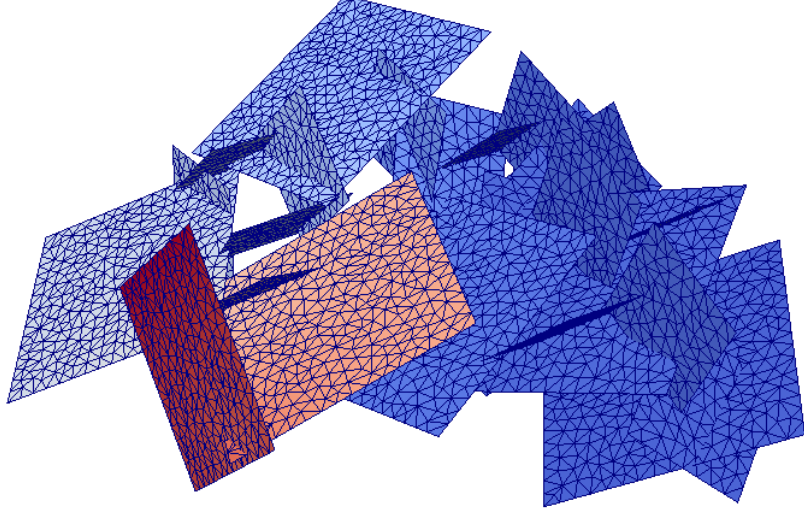


Figure 5.7: DFN 27: Spatial distribution of fractures for a DFN with 27 fractures

Table 5.1: DFN 27: Net flux in source (So) and sink (Si) fractures and flux mismatch Δ for various mesh sizes and VEM orders

Method	mesh 150			mesh 120		
	Si	So	Δ	Si	So	Δ
VEM-1	8.75	-8.22	0.53	8.70	-7.92	0.78
VEM-2	11.23	-9.78	1.45	11.16	-10.05	1.09
VEM-3	11.60	-10.36	1.23	11.64	-10.60	1.04
VEM-4	11.88	-10.76	1.12	11.89	-10.92	0.98
Method	mesh 90			mesh 60		
	Si	So	Δ	Si	So	Δ
VEM-1	9.01	-7.75	1.26	9.73	-8.32	1.42
VEM-2	11.18	-10.03	1.08	11.40	-10.26	1.14
VEM-3	11.64	-10.73	0.91	11.80	-10.89	0.9
VEM-4	11.91	-10.99	0.92	12.03	-11.17	0.86
Method	mesh 30			mesh 15		
	Si	So	Δ	Si	So	Δ
VEM-1	10.56	-8.51	2.05	10.71	-9.49	1.23
VEM-2	11.83	-10.77	1.06	11.91	-11.00	0.91
VEM-3	12.11	-11.25	0.86	12.13	-11.53	0.59
VEM-4	12.26	-11.48	0.78	10.21	-13.01	-2.81
Method	mesh 10			mesh 5		
	Si	So	Δ	Si	So	Δ
VEM-1	10.98	-9.18	1.81	11.36	-10.26	1.12
VEM-2	12.00	-11.09	0.90	12.12	-11.65	0.47

homogeneous Dirichlet boundary conditions on one edge of their boundary and homogeneous Neumann boundary conditions on the remaining edges. All other fractures have homogeneous Neumann boundary conditions and are therefore insulated on their boundaries. In absence of an exact solution, the difference Δ between the flux entering the system from F_2 (the source fracture), “So”, and the flux leaving it from F_1 (sink fracture), “Si”, is considered for assessing the quality of the obtained numerical approximation.

It should be noted that the methodology presented in this work does not guarantee nor aims to have local mass conservation in each fracture, since this is not explicitly imposed on any fracture. This means that the global mass conservation is well described, but the “local” flux balances (i.e., on each individual fracture) can be somewhat less accurate. On the other hand, these fracture flux balances are expected to improve with finer meshes as the method is converging to the solution. On the whole, the method can be seen as basically solving the DFN problem in one very complex 3D domain in space, that may however still be thought as a set of bidimensional domains.

Table 5.1 shows the net flux in the source and sink fractures, Si and So, respectively, as well as the difference Δ for mesh parameters (area of the largest element of the mesh) ranging from 5 to 150 and orders of the VEM space from 1 to 4. Only orders 1 and 2 are considered on the two finer meshes.

After extensive numerical experiments a trend emerged in the results; for order 1, convergence can be quite slow in the flux variable on these coarse meshes and displays oscillations, this can be attributed to the fact that the approximation of the flux is only piecewise constant and the projection of the VEM space functions for each element is onto a polynomial space of degree one, regardless of the number of edges of the element. Moving to higher order discretization spaces, the approximation of the flux improves. A marked improvement is obtained with second order VEM with respect to the first order, probably due to the piecewise linear structure of U . Further increasing the VEM order has a less noticeable effect, with practically no gain in moving to a third or fourth order

Table 5.2: Comparison of iterations for different choices of Q and preconditioner M^{-1}

Method/Area	Total DOF	Trace DOF	CG	Lumped	Dirichlet
			Iter	Iter	Iter
VEM-1/150	7209	2047	137	106	72
VEM-1/90	9220	2524	152	118	77
VEM-1/30	19116	4182	29891	138	80
VEM-1/5	75672	9833	NC	238	113
VEM-2/150	25028	3869	181	259	77
VEM-2/90	34038	4823	4537	286	74
VEM-2/30	79736	8139	NC	357	112

approximation. In addition, higher order discretizations might suffer from numerical instabilities due to very badly shaped elements. This is for example the case for the fourth order approximation on the mesh size 15, where instabilities cause a degenerate discrete solution as shown by the parameter Δ reported in Table 5.1. Further details on possible causes of instabilities are discussed later in Paragraph 5.4.

Remark 3. *When tackling a new DFN, a good practice would be to run it the first time with a coarse mesh and first order elements. The values of h and of u already provide a reliable indication of the order of magnitude of the correct solution, and using the flux values on each fracture one can establish a rule for selecting the fractures for which a mesh refinement is advisable. Fractures with less important contribution to the total flux through the DFN do not require a finer mesh. Afterwards, a new simulation can be launched with second order elements and the new adapted mesh.*

5.3. DFN - 120 fractures

We now consider a DFN consisting of 120 fractures, as shown in Figure 5.8. Dirichlet boundary conditions are imposed on a source and sink fracture whereas all other fractures have homogeneous Neumann boundary conditions. In Figure 5.9 we plot the solution for the sink fracture and for a selected fracture with insulated boundaries. As a comparison, results are shown for both the VEM approach of order 2 depicted in the present work and for the XFEM based optimization approach described in [18], starting from the same baseline mesh. A very good agreement between the solutions can be appreciated in the figure. Good agreement was also obtained for VEM of orders 1 and 3.

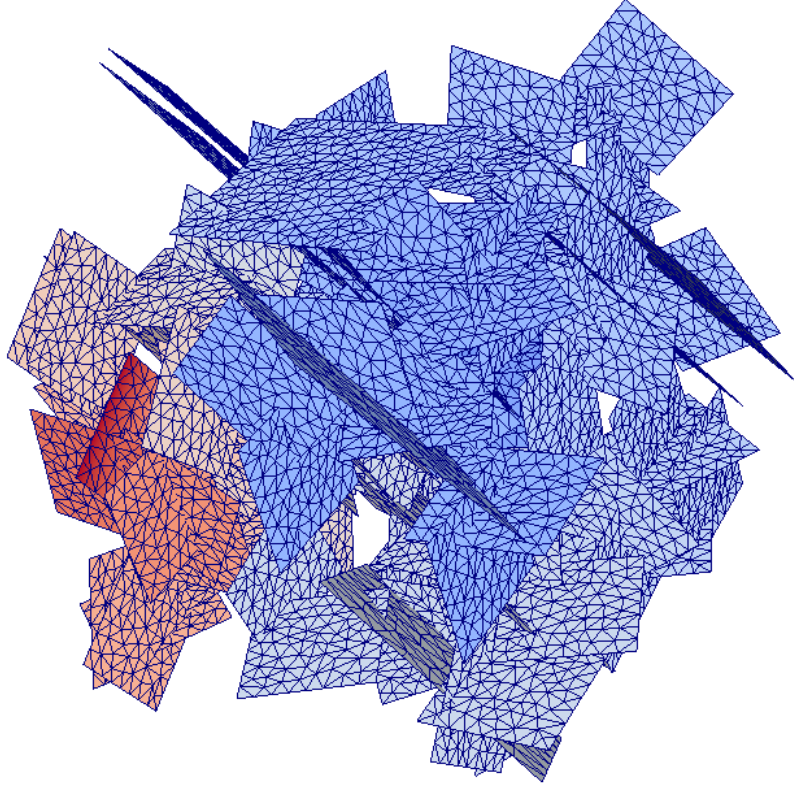


Figure 5.8: DFN 120: Spatial distribution of fractures for a DFN with 120 fractures

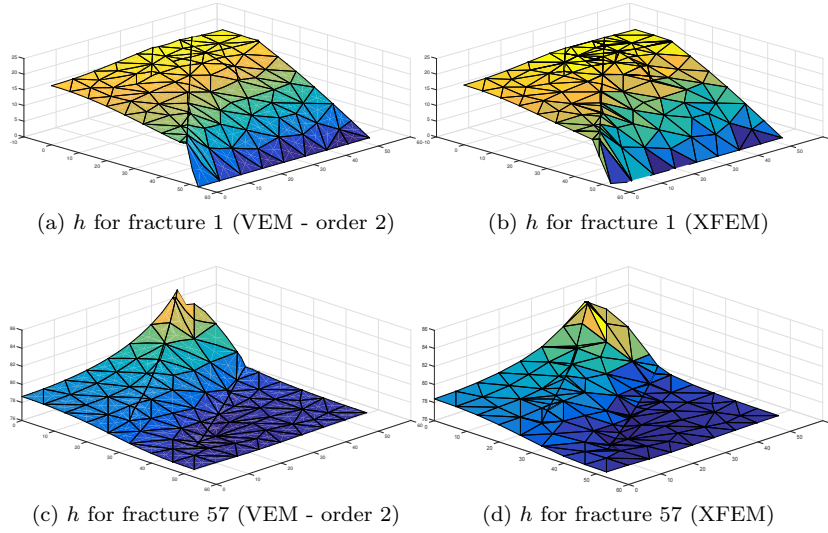


Figure 5.9: DFN 120: Large DFN comparison

423 In Table 5.2, we report the behaviour of 2 preconditioning techniques. Dif-
424 ferent mesh parameters and VEM of order 1 and 2 are considered. The table
425 displays the number of iterations required by the conjugate gradient (CG) rou-
426 tine compared to the performances of the preconditioned algorithm with the
427 Lumped and Dirichlet preconditioners. For the non preconditioned CG algo-
428 rithm, a rapid increase in the iteration number with mesh refinement can be
429 appreciated for both orders 1 and 2. As expected, the increase in iterations with
430 a preconditioner is much smaller, with the Dirichlet preconditioner performing
431 better than the Lumped preconditioner.

432 The notable improvement renders almost imperative the use of a precon-
433 ditioner, since the reduction in iteration number far outweighs the extra com-
434 putational cost that arises from the computation of the preconditioner. Cases
435 marked with NC stand for no convergence after 1 million iterations.

436 5.4. *A survey of troublesome situations*

437 In this subsection we describe some situations arisen in the simulations that
438 have proven to be difficult to handle numerically. The monomial basis for the
439 space of polynomials is notoriously bad conditioned, and the situation worsens
440 with increasing orders. We believe that this is the cause of the issues we are
441 presenting in this section, and they appear in elements with unsuitable shapes.
442 Some of these issues can be prevented if a mesh modifying procedure as men-
443 tioned in Remark 2 is used.

444 A first example is related to the DFN with 120 fractures, where a fracture
445 has two traces that are almost parallel and very close to each other, as in
446 Figure 5.10. This inevitably leads to elements with a bad aspect ratio, since
447 any attempt to obtain an adequate mesh would require a very large number of
448 small elements to fill the space between the two traces. The solution is stable up
449 to VEM of order 3, while when using a fourth order approximation the obtained
450 solution drastically changes (see Figure 5.11), and even falls below zero, which
451 is not compatible with the imposed boundary conditions, necessarily leading

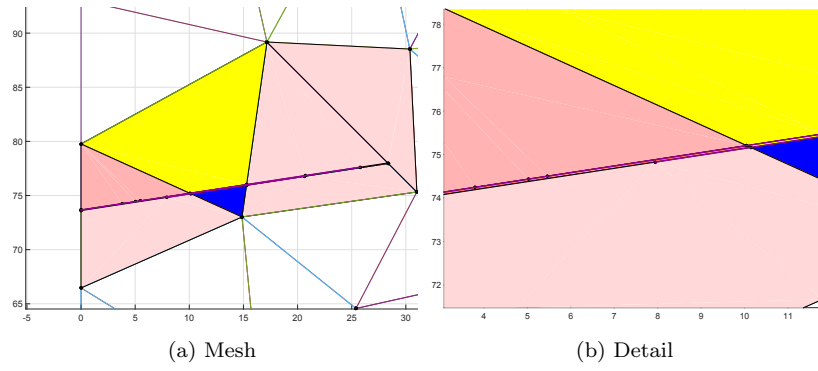


Figure 5.10: DFN 120: Detail of two very close and almost parallel traces

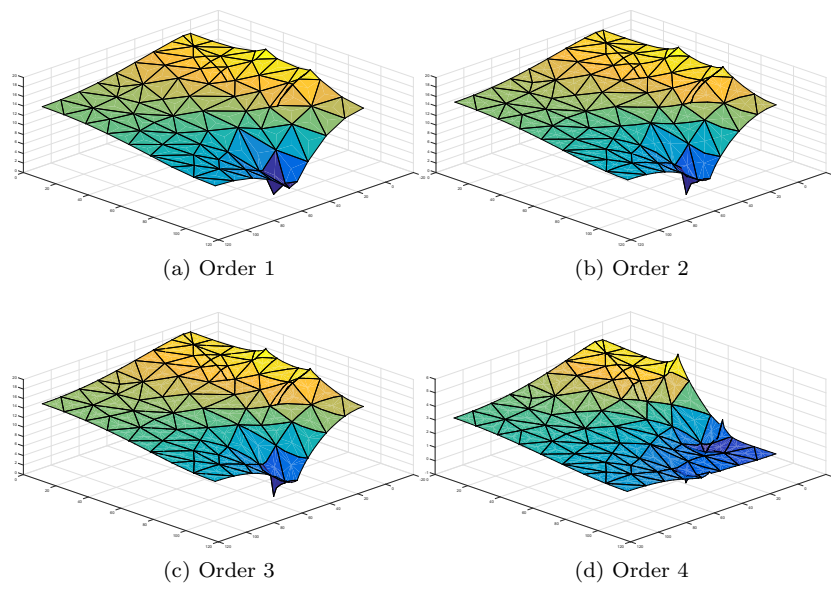


Figure 5.11: DFN 120: Comparison of results for problematic situations

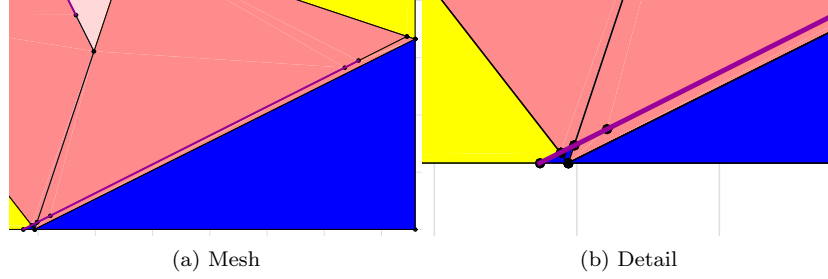


Figure 5.12: DFN 27: Detail of an unfortunate disposition of a mesh edge and a trace

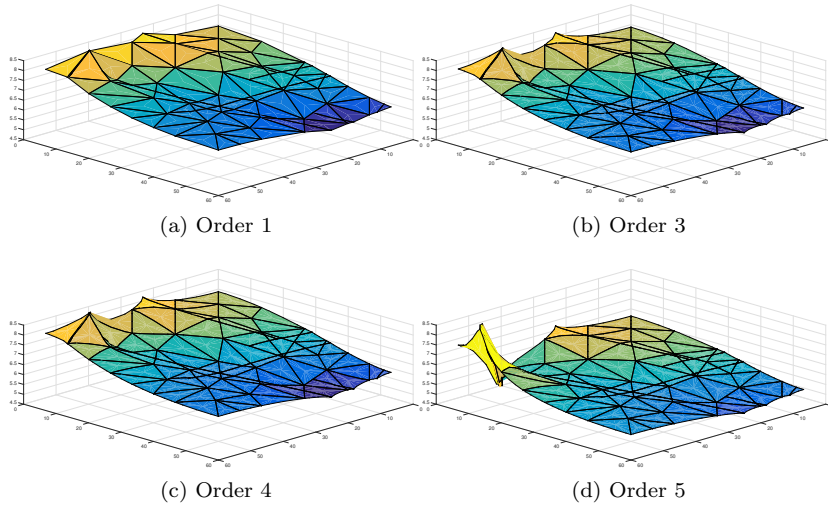


Figure 5.13: DFN 27: Comparison of results for problematic situations

to a solution bounded between 0 and 100. As a reference, one particularly problematic mesh element has an almost rectangular shape and an area of 0.58, with a length of 10.26 in one direction and 0.058 in the other (a 177 ratio). This is a degenerate octagon and for order 4 it has 38 DOFs (Figure 5.10). We remark that this particular configuration can be successfully dealt with VEM of orders from 1 to 3, and problems only appear with order 4 and higher.

A second documented problematic configuration, occurred on the DFN 27 problem, concerns badly shaped elements due not to the geometry of the DFN but to an unfortunate starting mesh, and is such that it may not be present with either a finer or a coarser mesh. This situation could be prevented applying the mesh smoothing process described in Remark 2. The situation is depicted in

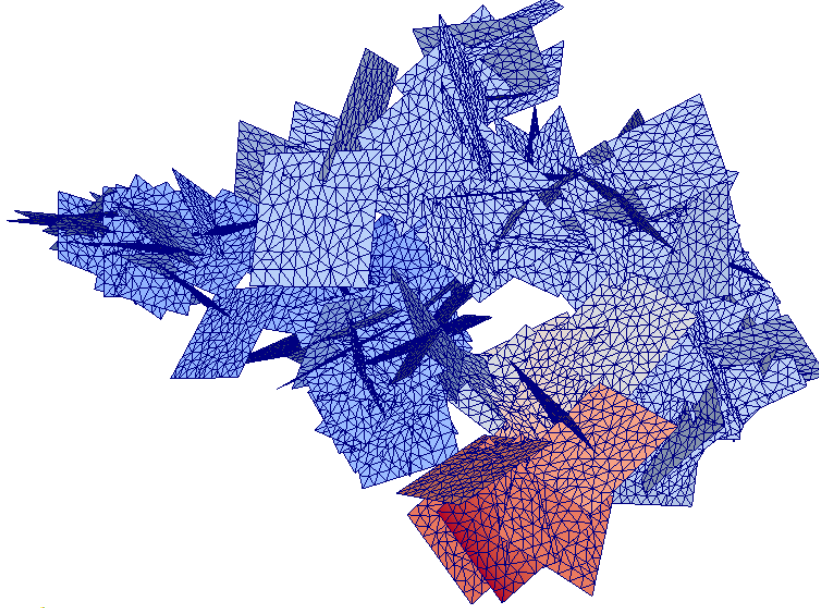


Figure 5.14: DFN 130: Spatial distribution of fractures for a DFN with 130 fractures

Figure 5.12, where we can see that the edge of an element is very close to a trace and has originated elements much more stretched in one direction than in the other. Furthermore, a very small element was generated next to the stretched element. The solution for VEM of order 5 becomes numerically unstable in this case, as shown by Figure 5.13. We remark that the major source of instability in this case is again the elongated element and not the neighboring small element.

Finally, we present the last case that is part of a medium size DFN with 130 fractures, shown in Figure 5.14, that includes parallel traces very close to each other, large disparity between trace lengths, highly heterogeneous element areas, element angles of less than 1 degree and complex trace intersections among other complications. More precisely, we have for the whole DFN that: minimum angle $= 0.41^\circ$, maximum trace length ≈ 45 , minimum trace length ≈ 0.01 and largest number of traces in a fracture $= 24$. An adequate globally conforming triangular mesh for this system would be quite difficult to obtain, if not impossible. With our approach, meshing can be done as usual (Figure 5.15) although it may lead to elements with undesirable shapes. It can be seen that irregularities in the

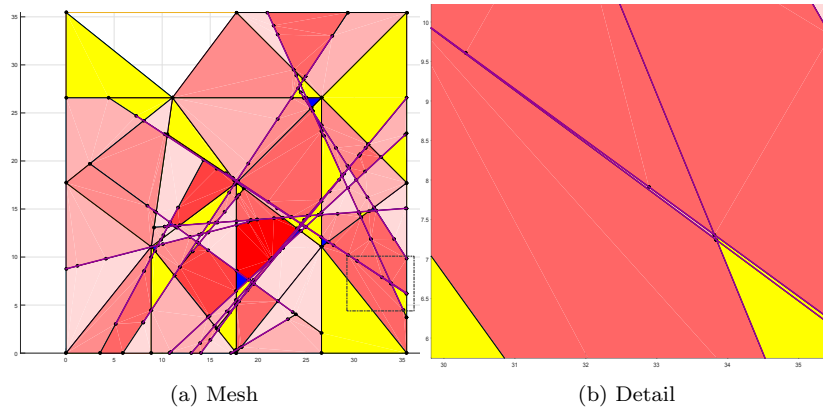


Figure 5.15: DFN 130: Detail of two traces meeting at a very small angle

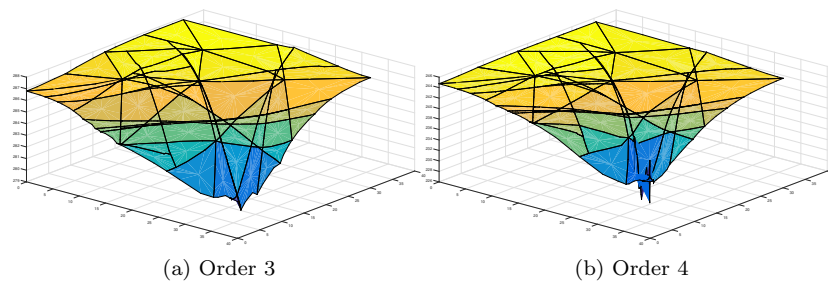


Figure 5.16: DFN 130: Comparison of results for problematic situations

479 solution were present only starting from VEM of order 4 approximations, again
 480 at a very elongated element between two traces meeting at a very small angle
 481 (Figure 5.16). The solution shows an uneven and rough behaviour that is further
 482 propagated to other fractures that have traces in common, and was not present
 483 in the solution obtained with the VEM of order 3.

484 6. Conclusions

485 In this work we have presented a novel method that constitutes a natural
 486 generalization of conforming Finite Elements for Discrete Fracture Network flow
 487 simulations. Local and global conformity is obtained using some of the features
 488 of the Virtual Element Method, and most importantly, global conformity is
 489 achieved without any constraints in the initial meshing process, that is per-
 490 formed independently for each fracture, nor any modification of DFN geometry.
 491 Convergence curves were presented as well as results for DFNs of small and
 492 medium scale, and the method has been shown to be robust enough to handle
 493 complex geometrical situations that arise in randomly generated DFNs.

494 After extensive numerical experiments, the following patterns were noticed:
 495 in general, all methods give a good approximation for the hydraulic head H ,
 496 and due to how the problem was implemented, continuity of H for the whole
 497 DFN is guaranteed. Even with VEM of order 1 the solutions are reliable for this
 498 variable, and this is due to the fact that we are using the primal formulation
 499 of the problem and the local conformity of the mesh allows for a more accurate
 500 representation of the jump of the derivative of H along the traces. In the case of
 501 the flux exchanged at the traces, U , the situation is different; only starting with
 502 a somewhat fine mesh can acceptable results be obtained for order 1. Order 2
 503 on the other hand, shows a marked improvement that can be attributed to the
 504 larger number of DOF but also to the improved approximation of the gradient
 505 of H and consequently of U . We remark that U is not obtained directly, but
 506 deriving the projection onto a polynomial space of the computed primal variable
 507 H .

Concerning the use of discretizations with increasing polynomial accuracy, for this application, we discourage going beyond order 2 based on the obtained results. Higher orders are not only less stable numerically on strongly distorted meshes, but also much more computationally expensive, and the improvement in accuracy is often not considerable. In fact, the exact solution of a DFN does not have in general high regularity and a cubic approximation of H and a quadratic approximation for U might be excessive. As we have seen however, whenever regularity is guaranteed, convergence for higher orders is as good as expected.

Simple FETI algorithms for domain decomposition were successfully implemented and show promise for possible parallelization of the resulting linear system. They prove to be nearly indispensable if a large system is to be solved due to the achievable reduction in the number of iterations required to solve the system.

Finally, much of the work done here in obtaining the globally conforming meshes as well as the idea for imposing matching conditions between corresponding degrees of freedom can be readily applied with few alterations to an implementation of a mixed formulation of the original problem using mixed Virtual Elements and will be the subject of future work.

Acknowledgements

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 through project *Tecniche numeriche per la simulazione di flussi in reti di fratture di grandi dimensioni*. The first author was also supported by the European Commission through the Erasmus Mundus Action 2-Strand1 ARCOIRIS programme, Politecnico di Torino.

534 **References**

- 535 [1] P. M. Adler, Fractures and Fracture Networks, Kluwer Academic, Dor-
536 drecht, 1999. doi:http://dx.doi.org/10.1007/978-94-017-1599-7.
- 537 [2] C. Fidelibus, G. Cammarata, M. Cravero, Hydraulic characterization of
538 fractured rocks. In: Abbie M, Bedford JS (eds) Rock mechanics: new re-
539 search., Nova Science Publishers Inc., New York, 2009.
- 540 [3] P. Panfili, A. Cominelli, Simulation of miscible gas injection in
541 a fractured carbonate reservoir using an embedded discrete frac-
542 ture model, in: Proceedings of Abu Dhabi International Petroleum
543 Exhibition and Conference, Society of Petroleum Engineers, 2014.
544 doi:http://dx.doi.org/10.2118/171830-MS.
- 545 [4] B. Decroux, O. Gosselin, Computation of effective dynamic properties of
546 naturally fractured reservoirs: Comparison and validation of methods, in:
547 EAGE Annual Conference & Exhibition incorporating SPE Europec, So-
548 ciety of Petroleum Engineers, 2013. doi:http://dx.doi.org/10.2118/164846-
549 MS.
- 550 [5] V. Martin, J. Jaffr, J. E. Roberts, Modeling fractures and barriers as inter-
551 faces for flow in porous media, SIAM Journal on Scientific Computing 26 (5)
552 (2005) 1667–1691. doi:http://dx.doi.org/10.1137/S1064827503429363.
- 553 [6] V. Lenti, C. Fidelibus, A BEM solution of steady-state flow
554 problems in discrete fracture networks with minimization of core
555 storage, Computers & Geosciences 29 (9) (2003) 1183 – 1190.
556 doi:http://dx.doi.org/10.1016/S0098-3004(03)00140-7.
- 557 [7] S. Li, Z. Xu, G. Ma, W. Yang, An adaptive mesh refinement method
558 for a medium with discrete fracture network: The enriched Persson’s
559 method, Finite Elements in Analysis and Design 86 (0) (2014) 41 – 50.
560 doi:http://dx.doi.org/10.1016/j.finel.2014.03.008.

- [8] T. Kalbacher, R. Mettier, C. McDermott, W. Wang, G. Kosakowski, T. Taniguchi, O. Kolditz, Geometric modelling and object-oriented software concepts applied to a heterogeneous fractured network from the Grimsel rock laboratory, *Comput. Geosci.* 11 (2007) 9–26. doi:<http://dx.doi.org/10.1007/s10596-006-9032-8>.
- [9] M. Vohralík, J. Maryška, O. Severýn, Mixed and nonconforming finite element methods on a system of polygons, *Applied Numerical Mathematics* 51 (2007) 176–193. doi:<http://dx.doi.org/10.1016/j.apnum.2006.02.005>.
- [10] 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:<http://dx.doi.org/10.1029/WR026i003p00479>.
- [11] 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:<http://dx.doi.org/10.1029/1999WR900118>.
- [12] J. Maryška, O. Severýn, M. Vohralík, Numerical simulation of fracture flow with a mixed-hybrid fem stochastic discrete fracture network model, *Computational Geosciences* 8 (3) (2005) 217–234. doi:<http://dx.doi.org/10.1007/s10596-005-0152-3>.
- [13] J. Erhel, J.-R. De Dreuzy, B. Poirriez, Flow simulation in three-dimensional discrete fracture networks, *SIAM Journal on Scientific Computing* 31 (4) (2009) 2688–2705. doi:<http://dx.doi.org/10.1137/080729244>.
- [14] G. Pichot, J. Erhel, J. de Dreuzy, A mixed hybrid mortar method for solving flow in discrete fracture networks, *Applicable Analysis* 89 (10) (2010) 1629–1643. doi:<http://dx.doi.org/10.1080/00036811.2010.495333>.

- [15] 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 (1) (2012) B86–B105. doi:http://dx.doi.org/10.1137/100804383.
- [16] 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:http://dx.doi.org/10.1137/120865884.
- [17] 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:http://dx.doi.org/10.1137/120882883.
- [18] 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:http://dx.doi.org/10.1016/j.jcp.2013.09.028.
- [19] S. Berrone, S. Pieraccini, S. Scial, F. Vicini, A parallel solver for large scale dfn flow simulations, *SIAM Journal on Scientific Computing* 37 (3) (2015) C285–C306. doi:http://dx.doi.org/10.1137/140984014.
- [20] M. F. Benedetto, S. Berrone, S. Pieraccini, S. Scialò, The virtual element method for discrete fracture network simulations, *Computer Methods in Applied Mechanics and Engineering* 280 (0) (2014) 135 – 156. doi:http://dx.doi.org/10.1016/j.cma.2014.07.016.
- [21] L. Beirão da Veiga, F. Brezzi, A. Cangiani, G. Manzini, L. D. Marini, A. Russo, Basic principles of virtual element methods, *Math. Models Methods Appl. Sci.* 23 (1) (2013) 199–214. doi:http://dx.doi.org/10.1142/S0218202512500492.
- [22] L. Beirão da Veiga, F. Brezzi, L. D. Marini, A. Russo, The hitchhikers guide to the virtual element method, *Mathematical Mod-*

- els and Methods in Applied Sciences 24 (08) (2014) 1541–1573.
doi:http://dx.doi.org/10.1142/S021820251440003X.
- [23] B. Ahmad, A. Alsaedi, F. Brezzi, L. D. Marini, A. Russo, Equivalent projectors for virtual element methods, Comput. Math. Appl. 66 (3) (2013) 376–391. doi:http://dx.doi.org/10.1016/j.camwa.2013.05.015.
- [24] A. Cangiani, G. Manzini, A. Russo, N. Sukumar, Hourglass stabilization and the virtual element method, International Journal for Numerical Methods in Engineering 102 (3-4) (2014) 404–436. doi:http://dx.doi.org/10.1002/nme.4854.
- [25] L. Beirão da Veiga, G. Manzini, A virtual element method with arbitrary regularity, IMA Journal of Numerical Analysis 34 (2) (2014) 759–781. doi:http://dx.doi.org/10.1093/imanum/drt018.
- [26] L. Beirão da Veiga, F. Brezzi, L. D. Marini, A. Russo, Virtual element methods for general second order elliptic problems on polygonal meshes, arXiv:1412.2646.
- [27] L. B. 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.
- [28] F. Brezzi, L. D. Marini, Virtual element methods for plate bending problems, Computer Methods in Applied Mechanics and Engineering 253 (0) (2013) 455 – 462. doi:http://dx.doi.org/10.1016/j.cma.2012.09.012.
- [29] P. F. Antonietti, L. B. 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:http://dx.doi.org/10.1137/13091141X.
- [30] L. B. da Veiga, K. Lipnikov, G. Manzini, The Mimetic Finite Difference

- 641 Method for Elliptic Problems, Springer International Publishing, 2014.
642 doi:http://dx.doi.org/10.1007/978-3-319-02663-3.
- 643 [31] K. Lipnikov, G. Manzini, M. Shashkov, Mimetic finite difference method,
644 Journal of Computational Physics 257, Part B (2014) 1163 – 1227.
645 doi:http://dx.doi.org/10.1016/j.jcp.2013.07.031.
- 646 [32] A. Klawonn, O. Widlund, FETI and Neumann-Neumann iterative
647 substructuring methods: Connections and new results, Comm. Pure
648 Appl. Math. 54 (1) (2001) 57–90. doi:http://dx.doi.org/10.1002/1097-
649 0312(200101)54:1<57::AID-CPA3>3.0.CO;2-D.
- 650 [33] J. Nocedal, S. J. Wright, Numerical Optimization, Springer, Berlin, 1999.
- 651 [34] A. Klawonn, FETI domain decomposition methods for second order elliptic
652 partial differential equations, GAMM-Mitteilungen 29 (2) (2006) 319–341.
653 doi:http://dx.doi.org/10.1002/gamm.201490036.
- 654 [35] V. Girault, R. Glowinski, Error analysis of a fictitious domain method
655 applied to a Dirichlet problem, Japan J. Indust. Appl. Math. 12 (3) (1995)
656 487–514. doi:http://dx.doi.org/10.1007/BF03167240.
- 657 [36] L. B. Wahlbin, Local behavior in finite element methods, in: Handbook
658 of numerical analysis, Vol. II, Handb. Numer. Anal., II, North-Holland,
659 Amsterdam, 1991, pp. 353–522. doi:http://dx.doi.org/10.1016/S1570-
660 8659(05)80040-7.