



Politecnico di Torino

Porto Institutional Repository

[Article] An optimization approach for large scale simulations of discrete fracture network flows

Original Citation:

Berrone S.; Pieraccini S.; Scialo' S. (2014). *An optimization approach for large scale simulations of discrete fracture network flows*. In: [JOURNAL OF COMPUTATIONAL PHYSICS](#), vol. 256 n. 1, pp. 838-853. - ISSN 0021-9991

Availability:

This version is available at : <http://porto.polito.it/2504656/> since: November 2012

Publisher:

Elsevier

Published version:

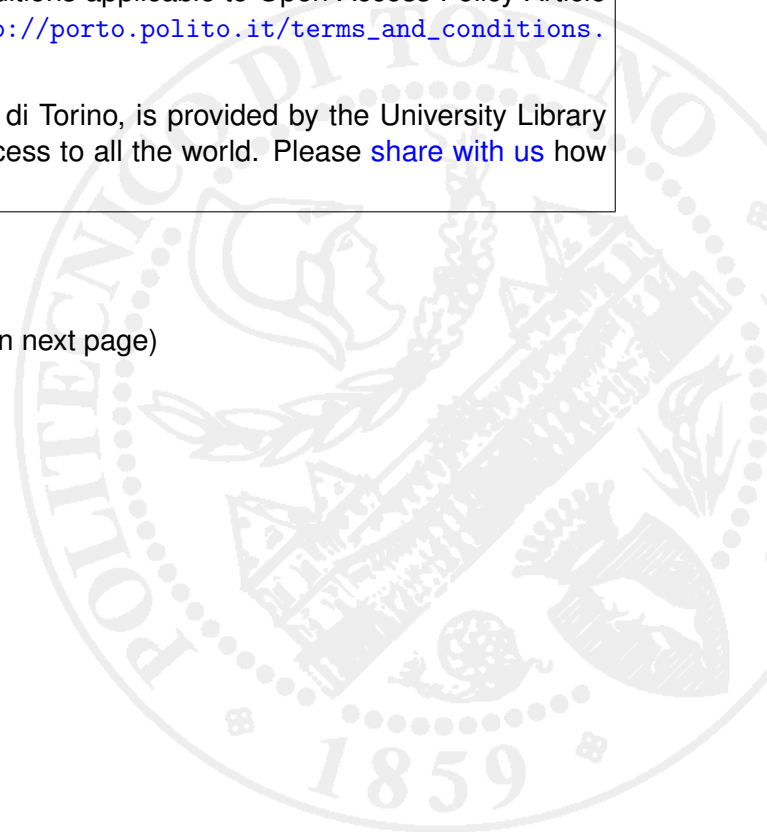
DOI:[10.1016/j.jcp.2013.09.028](https://doi.org/10.1016/j.jcp.2013.09.028)

Terms of use:

This article is made available under terms and conditions applicable to Open Access Policy Article ("Public - All rights reserved") , as described at http://porto.polito.it/terms_and_conditions.html

Porto, the institutional repository of the Politecnico di Torino, is provided by the University Library and the IT-Services. The aim is to enable open access to all the world. Please [share with us](#) how this access benefits you. Your story matters.

(Article begins on next page)



An optimization approach for large scale simulations of discrete fracture network flows

Stefano Berrone*, Sandra Pieraccini*, Stefano Scialò*

This is the authors' post-print version of an article published on *Journal of Computational Physics*, Volume 256 (2014), pp. 838-853, DOI:10.1016/j.jcp.2013.09.028.[†]

Abstract

In recent papers [1, 2] the authors introduced a new method for simulating subsurface flow in a system of fractures based on a PDE-constrained optimization reformulation, removing all difficulties related to mesh generation and providing an easily parallel approach to the problem. In this paper we further improve the method removing the constraint of having on each fracture a non empty portion of the boundary with Dirichlet boundary conditions. This way, Dirichlet boundary conditions are prescribed only on a possibly small portion of DFN boundary. The proposed generalization of the method in [1, 2] relies on a modified definition of control variables ensuring the non-singularity of the operator on each fracture. A conjugate gradient method is also introduced in order to speed up the minimization process.

Keywords: Fracture flows, Darcy flows, discrete fracture networks, optimization methods for elliptic problems, uncoupled large scale simulations

MSC subject classification: 65N30, 65N15, 65N50, 65J15

1 Introduction

Efficient numerical simulation of underground flow is of great interest in a large variety of practical applications, as for example enhanced oil/gas recovery, pollutant percolation and diffusion in aquifers, or carbon dioxide storage. The underground fluid flow is a multi-scale heterogeneous phenomenon, occurring in complex geological configurations usually characterized by networks of fractures surrounded by a porous rock matrix. The Discrete Fracture Network (DFN) approach models underground systems of fractures as 3D networks of intersecting discrete planar fractures. Diffusive phenomena in this system of fractures are governed by the Darcy law. At fracture intersections, called *traces*, mass balance and pressure continuity are preserved. The geological characteristics of the fractures, such as size, orientation, aspect ratio, density, permeability, are usually determined relying on stochastic data [3], and only probability distribution of data are usually available for a specific geological site. A huge number of numerical simulations is then necessary in order to perform sensitivity analysis

*Dipartimento di Scienze Matematiche, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129 Torino, Italy, stefano.berrone@polito.it, sandra.pieraccini@polito.it, stefano.scialo@polito.it

[†]This version does not contain journal formatting and may contain minor changes with respect to the published version. The final publication is available at <http://dx.doi.org/10.1016/j.jcp.2013.09.028>. The present version is accessible on PORTO, the Open Access Repository of Politecnico di Torino (<http://porto.polito.it>), in compliance with the Publisher's copyright policy as reported in the SHERPA-ROMEO website: <http://www.sherpa.ac.uk/romeo/issn/0021-9991/>

to the variability of the involved parameters. On the other hand, DFN simulations are very demanding from a computational point of view. Problem size is usually huge, involving a very large number of fractures. Moreover, for intricate fracture geometries, the generation of a good quality finite element triangulation conforming to the traces usually requires the introduction of many unknowns on each fracture, independently of the quality required for the numerical solution.

Many approaches are suggested in literature to circumvent these difficulties. A method based on a conforming mesh with mixed non-conforming finite elements is proposed in [4], while in other cases modifications of the geometry or of the mesh are introduced in order to preserve conformity and achieve a good quality mesh, such as in [5, 4] or in [6]. A different approach is suggested in [7], where the solution in the fractures is expressed as a function of the solution at the intersections. In other works it is suggested to rely on mortar methods to ease meshing procedure, as for example in [8, 9]: with this approach the mesh conformity constraint is relaxed but fracture meshes have to be aligned along the traces. In [10, 11, 12] the DFN is reduced into a system of mono-dimensional pipes connecting the traces with the surrounding fractures both preserving fracture topology and mitigating meshing related problems.

The present work further develops the approach introduced in [1, 2], in which the problem of the computation of the hydraulic head in a DFN is reformulated as a PDE-constrained optimization problem. The overall problem is split in a set of several independent sub-problems on each fracture of the system, coupled by the minimization of a proper functional. The use of Extended Finite Elements allows to capture the correct behaviour of the solution along traces even if grids are not conforming along fracture intersections and traces arbitrarily cut mesh elements. This way the meshes may be generated on each fracture in a completely independent way, disregarding fracture intersections and thus eliminating meshing difficulties.

Despite being applicable to very general DFN configurations, the formulation of the problem in the over-mentioned approach requires a non empty portion of Dirichelet boundary on each fracture of the system. In the present work a modification of the control variable and of the cost functional involved in the optimization problem is introduced, eliminating this constraint and allowing to prescribe Dirichelet boundary conditions only on (portion of) boundaries of a – possibly very small – subset of fractures. The use of a conjugate gradient method for the minimization process is also described. The behaviour of the method on fairly complex networks is shown through several numerical experiments.

The paper is organized as follows. In Section 2 we recall the physical model and the mathematical statement of the continuous problem introduced in [1, 2]. In Section 3 the PDE-constrained optimization problem is described along with the conjugate gradient algorithm used in the minimization process. Application of XFEM ideas to the DFN context is briefly accounted for in Section 4. In Section 5 we introduce the discrete version of the algorithm. Numerical experiments showing effectiveness of the method are reported and commented in Section 6.

2 Description of the problem

2.1 Problem formulation

Our target is the computation of the hydraulic head $H = \mathcal{P} + \zeta$ (being $\mathcal{P} = p/(\rho g)$ the pressure head, p the fluid pressure, g the gravitational acceleration constant, ρ the fluid density, ζ the elevation) in a DFN given by the union of a set of fractures. Let us model each fracture as an open planar polygonal set, F_i , with index i varying in a set \mathcal{J} . Let us also introduce on each fracture a 2D local coordinate system \hat{x}_i . Let Ω be the 3D set

$$\Omega = \bigcup_{i \in \mathcal{J}} F_i,$$

and $\partial\Omega$ the boundary of Ω , split as usual in a set $\Gamma_D \neq \emptyset$ with Dirichlet boundary conditions and a set Γ_N with Neumann boundary conditions, such that $\Gamma_D \cup \Gamma_N = \partial\Omega$ and $\Gamma_D \cap \Gamma_N = \emptyset$.

Note that the intersection of the closure of each couple of fractures is either an empty set or a set of non vanishing segments called *traces*, denoted by S_m , with index m varying in an index set \mathfrak{M} with cardinality $\#\mathfrak{M}$. For each fracture F_i , \mathcal{S}_i is the set of traces shared by F_i and other fractures while \mathcal{S} indicates the set of all the traces.

In the paper the following is assumed on the DFN: 1) $\bar{\Omega}$ is a connected set; 2) each trace S_m is shared by exactly two polygonal fractures F_i and F_j , $i \neq j$, such that $S_m \subseteq \bar{F}_i \cap \bar{F}_j$. The set of the two indices i and j of the fractures F_i and F_j sharing trace S_m is denoted by $I_{S_m} = \{i, j\}$, while for all $i \in \mathfrak{J}$, the subset $J_i \subset \mathfrak{J}$ contains indices of fractures with a non-empty intersection with F_i .

While referring the reader to [1] for more details, here we briefly recall the variational formulation of the problem. Let us define $\forall i \in \mathfrak{J}$ the following functional spaces:

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

and V_i' their dual spaces. The hydraulic head H_i in each fracture belongs to the space

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

where H_i^D is the restriction of the Dirichlet boundary condition $H|_{\Gamma_D} = H^D$ to $\Gamma_{iD} = \Gamma_D \cap \partial F_i$. In what follows Γ_{iD} can be an empty set, but $\Gamma_D = \bigcup_i \Gamma_{iD} \neq \emptyset$.

Let $\mathbf{K}_i(\hat{x}_i)$ be, for all $i \in \mathfrak{J}$, a symmetric and uniformly positive definite tensor called hydraulic conductivity tensor, which we assume dependent on the position and possibly different on each fracture. As documented in [1], the global hydraulic head H in the whole system Ω is obtained solving the following problems $\forall i \in \mathfrak{J}$, which model the diffusion of the hydraulic head on each fracture: find $H_i \in V_i^D$ such that $\forall v \in V_i$

$$\int_{F_i} \mathbf{K}_i \nabla H_i \nabla v d\Omega = \int_{F_i} q_i v d\Omega + \int_{\Gamma_N \cap \partial F_i} G_i^N v|_S d\Gamma + \sum_{S \in \mathcal{S}_i} \int_S \left[\left[\frac{\partial H_i}{\partial \hat{\nu}_S^i} \right] \right]_S v|_S d\Gamma, \quad (1)$$

where G_i^N is the restriction to $\Gamma_{iN} = \Gamma_N \cap \partial F_i$ of the Neumann boundary condition G^N imposed on Γ_N . The quantity $\frac{\partial H_i}{\partial \hat{\nu}_S^i} = (\hat{n}_S^i)^T \mathbf{K}_i \nabla H_i$ is the outward co-normal derivative of the hydraulic head, being \hat{n}_S^i the unit vector normal to the trace S . The symbol $\left[\left[\frac{\partial H_i}{\partial \hat{\nu}_S^i} \right] \right]_S$ denotes the jump of the co-normal derivative along \hat{n}_S^i , being this jump independent of the orientation of \hat{n}_S^i . According to (1), the diffusion of H_i is contributed by the following terms: the external load in each fracture (first term of the right hand side); the Neumann boundary conditions (second term); the net flow of hydraulic head entering in the fracture at each trace (last term).

Equations (1) are coupled by the following matching conditions, which prescribe global continuity of the hydraulic head and conservation of hydraulic fluxes across each trace S_m , $m \in \mathfrak{M}$:

$$H_i|_{S_m} - H_j|_{S_m} = 0, \quad \text{for } i, j \in I_{S_m}, \quad (2)$$

$$\left[\left[\frac{\partial H_i}{\partial \hat{\nu}_{S_m}^i} \right] \right]_{S_m} + \left[\left[\frac{\partial H_j}{\partial \hat{\nu}_{S_m}^j} \right] \right]_{S_m} = 0, \quad \text{for } i, j \in I_{S_m}. \quad (3)$$

Note that due to condition (2) the hydraulic head H on the whole domain Ω belongs to the space

$$V^D = H_D^1(\Omega) = \left\{ v \in \prod_{i \in \mathfrak{J}} V_i^D : (v|_{F_i})|_{S_m} = (v|_{F_j})|_{S_m}, \quad i, j \in I_{S_m}, \quad \forall m \in \mathfrak{M} \right\}. \quad (4)$$

3 Optimization approach

Following the approach described in [1], instead of solving the coupled differential problems on the fractures (1) $\forall i \in \mathfrak{I}$ with the corresponding matching conditions (2), (3), we introduce a PDE-constrained optimization problem. In order to ease notation and for a concise and clear description, in the following of this Section we assume that the traces $S \in \mathcal{S}$ are disjoint, recalling that as stated in [1], this assumption can be dropped replacing occurrences of each single trace S with the union of connected traces. Further, in our discrete formulation the assumption naturally drops thanks to the choice of the functional spaces (see again [1]). Let us introduce for each trace $S \in \mathcal{S}$ a suitable space \mathcal{U}^S and its dual $(\mathcal{U}^S)'$. Similar spaces are introduced on the set of traces belonging to a fracture $F_i, \forall i \in \mathfrak{I}$, and on the full set of traces \mathcal{S} :

$$\mathcal{U}^{S_i} = \prod_{S \in \mathcal{S}_i} \mathcal{U}^S, \quad \mathcal{U} = \prod_{i \in \mathfrak{I}} \mathcal{U}^{S_i}.$$

Now, let us fix a trace S and let $S \subseteq \bar{F}_i \cap \bar{F}_j$. We introduce suitable variables $U_i^S, U_j^S \in \mathcal{U}^S$ which will act as control variables, defined as $U_i^S = \alpha H_{i|_S} + \left[\frac{\partial H_i}{\partial \nu_i^S} \right]_S$ and $U_j^S = \alpha H_{j|_S} + \left[\frac{\partial H_j}{\partial \nu_j^S} \right]_S$ respectively, where α is a positive fixed parameter. This generalizes the approach proposed in [1] where U_i^S is set equal to flux jump, thus allowing $\Gamma_{iD} = \emptyset$ on possibly all but one fractures. We set

$$U_i = \prod_{S \in \mathcal{S}_i} U_i^S \in \mathcal{U}^{S_i}, \quad U = \prod_{i \in \mathfrak{I}} U_i \in \mathcal{U},$$

i.e. U_i is the tuple of functions U_i^S with $S \in \mathcal{S}_i$, and U is the $2(\#\mathfrak{M})$ -tuple of control functions on all traces in $\bar{\Omega}$.

We also introduce the Riesz isomorphisms $\Lambda_{\mathcal{U}^S} : \mathcal{U}^S \rightarrow \mathcal{U}^{S'}$, $\Lambda_{\mathcal{U}^{S_i}} : \mathcal{U}^{S_i} \rightarrow \mathcal{U}^{S_i'}$ and $\Lambda_{\mathcal{U}} : \mathcal{U} \rightarrow \mathcal{U}'$ and the following linear bounded operators and their duals:

$$\begin{aligned} A_i &\in \mathcal{L}(V_i, V_i'), & \langle A_i H_i^0, v \rangle_{V_i', V_i} &= (\mathbf{K} \nabla H_i^0, \nabla v) + \alpha \left(H_i^0|_{\mathcal{S}_i}, v|_{\mathcal{S}_i} \right)_{\mathcal{S}_i}, \\ A_i^D &\in \mathcal{L}(V_i^D, V_i'), & \langle A_i^D \mathcal{R}_i H_i^D, v \rangle_{V_i', V_i} &= (\mathbf{K} \nabla \mathcal{R}_i H_i^D, \nabla v) + \alpha \left((\mathcal{R}_i H_i^D)|_{\mathcal{S}_i}, v|_{\mathcal{S}_i} \right)_{\mathcal{S}_i}, \\ B_i^S &\in \mathcal{L}(\mathcal{U}^S, V_i'), & \langle B_i^S U_i, v \rangle_{V_i', V_i} &= \langle U_i^S, v|_{\mathcal{S}_i} \rangle_{\mathcal{U}^S, \mathcal{U}^{S'}}, \\ B_i &= \prod_{S \in \mathcal{S}_i} B_i^S \in \mathcal{L}(\mathcal{U}^{S_i}, V_i'), & \langle B_i U_i, v \rangle_{V_i', V_i} &= \langle U_i, v|_{\mathcal{S}_i} \rangle_{\mathcal{U}^{S_i}, \mathcal{U}^{S_i'}}, \end{aligned}$$

with $H_i^0 \in V_i$, $H_i^D \in V_i^D$, $v \in V_i$, and the operator \mathcal{R}_i is the lifting of the Dirichlet boundary conditions on Γ_{iD} if not empty. Dual operators are $A_i^* \in \mathcal{L}(V_i, V_i')$,

$$C_i^S = (B_i^S)^* \in \mathcal{L}(V_i, \mathcal{U}^{S'}), \quad C_i = (B_i)^* \in \mathcal{L}(V_i, \mathcal{U}^{S_i'}).$$

The operator $B_{iN} \in \mathcal{L}(H^{-\frac{1}{2}}(\Gamma_{iN}), V_i')$ imposing Neumann boundary conditions is defined such that

$$\langle B_{iN} G_i^N, v \rangle_{V_i', V_i} = \langle G_i^N, v|_{\Gamma_{iN}} \rangle_{H^{-\frac{1}{2}}(\Gamma_{iN}), H^{\frac{1}{2}}(\Gamma_{iN})} = \left\langle \frac{\partial H_i}{\partial \hat{\nu}_{\Gamma_{iN}}}, v|_{\Gamma_{iN}} \right\rangle_{H^{-\frac{1}{2}}(\Gamma_{iN}), H^{\frac{1}{2}}(\Gamma_{iN})}.$$

Problems (1) can now be written as follows: $\forall i \in \mathfrak{I}$, find $H_i \in V_i^D$, with $H_i = H_i^0 + \mathcal{R}_i H_i^D$ and $H_i^0 \in V_i$, such that

$$A_i H_i^0 = q_i + B_i U_i + B_{iN} G_i^N - A_i^D \mathcal{R}_i H_i^D, \quad \text{in } F_i. \quad (5)$$

We remark that, if $\alpha > 0$, the solution H_i to (5) exists and is unique for a non isolated fracture even if we set Neumann boundary conditions on the whole ∂F_i .

We can now define the differentiable functional $J : \mathcal{U} \rightarrow \mathbb{R}$ as

$$\begin{aligned}
J(U) &= \sum_{S \in \mathcal{S}} J^S(U) \\
&= \sum_{S \in \mathcal{S}} \left(\|C_i^S H_i(U_i) - C_j^S H_j(U_j)\|_{\mathcal{U}^{S'}}^2 + \|U_i^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} C_i^S H_i(U_i) + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} C_j^S H_j(U_j)\|_{\mathcal{U}^S}^2 \right) \\
&= \frac{1}{2} \sum_{i \in I} \sum_{S \in \mathcal{S}_i} \left(\|C_i^S H_i(U_i) - C_j^S H_j(U_j)\|_{\mathcal{U}^{S'}}^2 + \|U_i^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} C_i^S H_i(U_i) + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} C_j^S H_j(U_j)\|_{\mathcal{U}^S}^2 \right) \\
&= \frac{1}{2} \sum_{i \in I} \left\| \prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) - C_j^S H_j(U_j)) \right\|_{\mathcal{U}^{S'}}^2 + \frac{1}{2} \sum_{i \in I} \|U_i + \prod_{S \in \mathcal{S}_i} U_j^S - \alpha \Lambda_{\mathcal{U}^{S_i}}^{-1} \prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) + C_j^S H_j(U_j))\|_{\mathcal{U}^{S_i}}^2 \quad (6)
\end{aligned}$$

where quantity $\prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) \pm C_j^S H_j(U_j))$ denotes the tuple of functions $(C_i^S H_i(U_i) \pm C_j^S H_j(U_j))$ with $S \in \mathcal{S}_i$, and $i, j \in I_S$. Moreover $H_\ell(U_\ell)$ denotes the solution of (5) corresponding to the control variable U_ℓ , $\ell = i, j$.

Proposition 1. *Setting $\mathcal{U}^S = \mathbf{H}^{-\frac{1}{2}}(S)$ and letting $C_i^S \in \mathcal{L}(V_i, \mathbf{H}^{\frac{1}{2}}(S))$ be the trace operator, there exists a unique control variable U vanishing the functional $J(U)$ and a corresponding unique solution H satisfying problems (5) $\forall i \in \mathcal{I}$ that is solution to (1)-(3).*

Proof. We sketch very briefly the proof as it follows from classical arguments. Resorting to the classical formulation of the problem on sub-fractures as recalled in [1], it can be proven that exists a unique solution $H \in V^D$ for the hydraulic head on the DFN satisfying (1), $\forall i \in \mathcal{I}$, and (2), (3), $\forall m \in \mathfrak{M}$, that are trivially equivalent to (5), $\forall i \in \mathcal{I}$, and to

$$H_{i|S_m} - H_{j|S_m} = 0, \quad U_i^S - \alpha H_{i|S} + U_j^S - \alpha H_{j|S} = 0, \quad \text{for } i, j \in I_{S_m}, \quad \forall m \in \mathfrak{M}. \quad (7)$$

As in [1], since the vanishing of the two terms of the functional J is equivalent to the imposition of the matching conditions (7), the thesis follows. \square

Based on previous discussion, problems (5) coupled with (7) are replaced by the following optimization problem:

$$\min J(U) \quad \text{subject to (5), } \forall i \in \mathcal{I}. \quad (8)$$

In the following result we state optimality conditions for (8).

Proposition 2. *The optimal control $U \in \mathcal{U}$ satisfying (8) is given by the system of conditions (5) and*

$$B_i^* P_i + \Lambda_{\mathcal{U}^{S_i}} \left(U_i + \prod_{S \in \mathcal{S}_i} U_j^S \right) - \alpha \prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) + C_j^S H_j(U_j)) = 0, \quad (9)$$

$\forall i \in \mathcal{I}$, where the functions $P_i \in V_i$ are the solution of equation

$$A_i^* P_i = C_i^* \Lambda_{\mathcal{U}^{S_i}}^{-1} \left[\prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) - C_j^S H_j(U_j)) + \alpha^2 \prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) + C_j^S H_j(U_j)) \right] - \alpha C_i^* \left(U_i + \prod_{S \in \mathcal{S}_i} U_j^S \right), \quad \text{in } F_i, \quad (10)$$

in which homogeneous Dirichlet and Neumann boundary conditions on Γ_{iD} and Γ_{iN} , respectively, are prescribed.

Proof. Let us differentiate the cost functional with respect to the control variable U_i :

$$\begin{aligned}
J'(U)(v_i - U_i) &= \sum_{S \in \mathcal{S}_i} J^{S'}(U_i)(v_i - U_i) \\
&= 2 \sum_{S \in \mathcal{S}_i} \left[(C_i^S H_i(U_i) - C_j^S H_j(U_j), C_i^S (H_i(v_i) - H_i(U_i)))_{\mathcal{U}^{S'}} \right. \\
&\quad \left. + \left(U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} (C_i^S H_i(U_i) + C_j^S H_j(U_j)), v_i^S - U_i^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} (C_i^S H_i(v_i) - C_i^S H_i(U_i)) \right)_{\mathcal{U}^S} \right] \\
&= 2 \left\langle C_i^* \Lambda_{\mathcal{U}^{S_i}}^{-1} \prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) - C_j^S H_j(U_j)), H_i(v_i) - H_i(U_i) \right\rangle_{V_i', V_i} \\
&\quad + 2 \left\langle \Lambda_{\mathcal{U}^{S_i}} \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} (C_i^S H_i(U_i) + C_j^S H_j(U_j))), v_i - U_i \right\rangle_{\mathcal{U}^{S_i'}, \mathcal{U}^{S_i}} \\
&\quad - 2\alpha \left\langle C_i^* \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} (C_i^S H_i(U_i) + C_j^S H_j(U_j))), H_i(v_i) - H_i(U_i) \right\rangle_{V_i', V_i} \\
&= 2 \left\langle A_i^* P_i, A_i^{-1} B_i (v_i - U_i) \right\rangle_{V_i', V_i} + 2 \left\langle \Lambda_{\mathcal{U}^{S_i}} \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} (C_i^S H_i(U_i) + C_j^S H_j(U_j))), v_i - U_i \right\rangle_{\mathcal{U}^{S_i'}, \mathcal{U}^{S_i}} \\
&= 2 \left\langle B_i^* P_i + \Lambda_{\mathcal{U}^{S_i}} \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} (C_i^S H_i(U_i) + C_j^S H_j(U_j))), v_i - U_i \right\rangle_{\mathcal{U}^{S_i'}, \mathcal{U}^{S_i}}.
\end{aligned}$$

Thus, the vanishing of this last term yields (9). \square

Instead of solving equations (5), (9), (10), we set up a minimization process for problem (8). This is organized in such a way that only the decoupled solution of the local problems (5) is needed. Here we use the Fletcher and Reeves conjugate gradient method [13]. Let us denote by $\nabla J(U_i)$ the Frechet derivative of the functional J with respect to the control variables on the fracture F_i , $\forall i \in \mathcal{I}$, and by $\nabla J(U)$ the whole derivative:

$$\nabla J(U_i) = B_i^* P_i + \Lambda_{\mathcal{U}^{S_i}} \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1} (C_i^S H_i(U_i) + C_j^S H_j(U_j))), \quad (11)$$

$$\nabla J(U) = \prod_{i \in \mathcal{I}} \nabla J(U_i). \quad (12)$$

The method used is depicted in the following algorithm.

Algorithm 3.

1. Set $k = 0$ and initial guess for control variable U^0 ;
 2. find $H^0 = H(U^0)$ solving on each fracture the primal problem (5);
 3. find $P(U^0)$ solving on each fracture the dual problem (10);
 4. evaluate $\nabla J(U^0)$ by equation (12);
 5. set $(\delta U)^0 = -\Lambda_{\mathcal{U}}^{-1} \nabla J(U^0)$;
 6. While $J(U^k) \neq 0$ do:
 - 6.1. choose a step-size λ^k along direction $(\delta U)^k$;
 - 6.2. set $U^{k+1} = U^k + \lambda^k (\delta U)^k$;
 - 6.3. $\forall i \in \mathcal{I}$ solve primal problem (5) to find $H_i(U^{k+1})$;
 - 6.4. $\forall i \in \mathcal{I}$ solve dual problem (10) to find $P_i(U^{k+1})$;
 - 6.5. evaluate $\nabla J(U^{k+1})$ by (12);
 - 6.6. set $\beta^{k+1} = \|\nabla J(U^{k+1})\|_{\mathcal{U}'}^2 / \|\nabla J(U^k)\|_{\mathcal{U}'}^2$;
 - 6.7. set $(\delta U)^{k+1} = -\Lambda_{\mathcal{U}}^{-1} \nabla J(U^{k+1}) + \beta^{k+1} \delta U^k$;
 - 6.8. $k = k + 1$;
- end do.

Let us evaluate the optimal step-size λ which can be used in the previous algorithm at steps 6.1-6.2. Given a variation δU_i for the control variable on each fracture F_i and $\delta U = \sum_{i \in \mathcal{I}} \delta U_i$, let $\delta H_i \in V_i, \forall i \in \mathcal{I}$, be defined as the solution of the problem

$$A_i \delta H_i = B_i \delta U_i, \quad \text{in } F_i, \quad (13)$$

corresponding to homogeneous Dirichlet and Neumann boundary conditions on Γ_{iD} (if non-empty) and Γ_{iN} , respectively.

Proposition 4. *Let us increment the control variable U of a step $\lambda \delta U$, the optimal step-size λ is*

$$\lambda = -\langle \nabla J(U), \delta U \rangle_{\mathcal{U}', \mathcal{U}} \left\{ \sum_{S \in \mathcal{S}} (\|C_i^S \delta H_i - C_j^S \delta H_j\|_{\mathcal{U}^{S'}}^2 + \|\delta U_i^S + \delta U_j^S\|_{\mathcal{U}^S}^2 + \alpha^2 \|C_i^S \delta H_i + C_j^S \delta H_j\|_{\mathcal{U}^{S'}}^2) - 2\alpha \sum_{i \in \mathcal{I}} \left(\prod_{S \in \mathcal{S}_i} (\delta U_i^S + \delta U_j^S), \Lambda_{\mathcal{U}^{S_i}}^{-1} C_i \delta H_i \right)_{\mathcal{U}^{S_i}} \right\}^{-1}. \quad (14)$$

Proof. We have

$$\begin{aligned}
J(U + \lambda\delta U) &= \sum_{S \in \mathcal{S}} \|C_i^S H_i(U_i) - C_j^S H_j(U_j) + \lambda(C_i^S \delta H_i - C_j^S \delta H_j)\|_{\mathcal{U}^{S'}}^2 \\
&+ \sum_{S \in \mathcal{S}} \|U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j)) + \lambda(\delta U_i^S + \delta U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S \delta H_i + C_j^S \delta H_j))\|_{\mathcal{U}^S}^2 \\
&= J(U) + 2\lambda \sum_{i \in I} \sum_{S \in \mathcal{S}_i} \left((C_i^S H_i(U_i) - C_j^S H_j(U_j), C_i^S \delta H_i)_{\mathcal{U}^{S'}} + (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j)), \delta U_i^S)_{\mathcal{U}^S} \right) \\
&- \alpha \left(U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j)), \Lambda_{\mathcal{U}^S}^{-1} C_i^S \delta H_i \right)_{\mathcal{U}^S} - 2\lambda^2 \alpha \sum_{i \in I} \sum_{S \in \mathcal{S}_i} \left(\delta U_i^S + \delta U_j^S, \Lambda_{\mathcal{U}^S}^{-1} C_i^S \delta H_i \right)_{\mathcal{U}^S} \\
&+ \lambda^2 \sum_{S \in \mathcal{S}} (\|C_i^S \delta H_i - C_j^S \delta H_j\|_{\mathcal{U}^{S'}}^2 + \|\delta U_i^S + \delta U_j^S\|_{\mathcal{U}^S}^2 + \alpha^2 \|C_i^S \delta H_i + C_j^S \delta H_j\|_{\mathcal{U}^{S'}}^2)
\end{aligned}$$

Moreover,

$$\begin{aligned}
&J(U + \lambda\delta U) - J(U) + 2\lambda^2 \alpha \sum_{i \in I} \left(\prod_{S \in \mathcal{S}_i} (\delta U_i^S + \delta U_j^S), \Lambda_{\mathcal{U}^S}^{-1} C_i^S \delta H_i \right)_{\mathcal{U}^S} \\
&- \lambda^2 \sum_{S \in \mathcal{S}} (\|C_i^S \delta H_i - C_j^S \delta H_j\|_{\mathcal{U}^{S'}}^2 + \|\delta U_i^S + \delta U_j^S\|_{\mathcal{U}^S}^2 + \alpha^2 \|C_i^S \delta H_i + C_j^S \delta H_j\|_{\mathcal{U}^{S'}}^2) \\
&= 2\lambda \sum_{i \in I} \left(\left(\prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) - C_j^S H_j(U_j), C_i^S \delta H_i) \right)_{\mathcal{U}^{S'}} + \left(\prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j))), \delta U_i^S \right)_{\mathcal{U}^S} \right) \\
&- \alpha \left(\prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j))), \Lambda_{\mathcal{U}^S}^{-1} C_i^S \delta H_i \right)_{\mathcal{U}^S} \\
&= 2\lambda \sum_{i \in I} \left(\left\langle C_i^* \Lambda_{\mathcal{U}^{S_i}}^{-1} \prod_{S \in \mathcal{S}_i} (C_i^S H_i(U_i) - C_j^S H_j(U_j)), \delta H_i \right\rangle_{V_i', V_i} \right. \\
&- \alpha \left\langle C_i^* \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j))), \delta H_i \right\rangle_{V_i', V_i} \\
&\left. + \left\langle \Lambda_{\mathcal{U}^{S_i}} \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j))), \delta U_i \right\rangle_{\mathcal{U}^{S_i'}, \mathcal{U}^{S_i}} \right) \\
&= 2\lambda \sum_{i \in I} \langle A_i^* P_i, A_i^{-1} B_i \delta U_i \rangle_{V_i', V_i} + 2\lambda \sum_{i \in I} \left\langle \Lambda_{\mathcal{U}^{S_i}} \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j))), \delta U_i \right\rangle_{\mathcal{U}^{S_i'}, \mathcal{U}^{S_i}} \\
&= 2\lambda \sum_{i \in I} \left\langle B_i^* P_i + \Lambda_{\mathcal{U}^{S_i}} \prod_{S \in \mathcal{S}_i} (U_i^S + U_j^S - \alpha \Lambda_{\mathcal{U}^S}^{-1}(C_i^S H_i(U_i) + C_j^S H_j(U_j))), \delta U_i \right\rangle_{\mathcal{U}^{S_i'}, \mathcal{U}^{S_i}}
\end{aligned}$$

Then, deriving $\mathcal{J}(\lambda) := J(U + \lambda\delta U)$ with respect to λ and vanishing this derivative, we get (14). \square

4 The Extended Finite Element Method in the DFN context

In this section we briefly describe a discretization approach via extended finite elements for DFN problems that allows us to build the numerical triangulation independently of the traces disposition on each fracture. The

solution to Problem (1) with matching conditions (2)-(3) is in general a continuous function with discontinuous gradient along traces. A numerical solution based on standard Finite Elements (FE) would require the triangulation to be conforming to the traces, this in turn requiring a coupling in the meshing process for all the fractures in the system. The Extended Finite Element Method (XFEM) [14, 15, 16, 17], instead, introduces in the FE approximation spaces additional basis functions, called enrichment basis functions, in order to reproduce the irregular behaviour of the solution independently of the mesh. For a detailed description of the XFEM approach we refer the interested reader to the cited references. Let us first consider for simplicity a single trace S on a fixed fracture F . Let V_δ^{fem} be the standard FE trial and test spaces defined on the elements of a triangulation on F non conforming to the trace and spanned by Lagrangian basis functions ϕ_k , for k ranging in an index set \mathcal{I} . Let Φ be a function well approximating the irregular behaviour of H in a neighbourhood of the trace S . Starting from Φ and basis functions ϕ_k , using the Partition of Unity Method [18], new basis functions are introduced into the space V_δ^{fem} , enriching its approximation capabilities. The additional basis functions are clearly required only in the elements of the triangulation which are intersected by the trace. In this way the irregular behaviour of the numerical solution is determined by the enrichment functions introduced, and is independent of the position of elements with respect to the trace. The XFEM space is then:

$$V_\delta^{xfem} = \text{span} \left(\{\phi_k\}_{k \in \mathcal{I}}, \{\phi_k \Phi\}_{k \in \mathcal{J}} \right)$$

where $\mathcal{J} \subset \mathcal{I}$ is the subset of the degrees of freedom involved in the enrichment. Consequently the approximate solution with the XFEM will have the following structure:

$$h_\delta^{xfem} = \sum_{k \in \mathcal{I}} h_k^{xfem} \phi_k + \sum_{k \in \mathcal{J}} a_k^{xfem} \phi_k \Phi,$$

where h_k^{xfem} and a_k^{xfem} are the unknowns related to the standard and enriching basis functions, respectively.

If more traces are present on the fracture F , the approach is simply generalized as follows: the XFEM space is taken as

$$V_\delta^{xfem} = \text{span} \left(\{\phi_k\}_{k \in \mathcal{I}}, \cup_{m \in \mathfrak{M}_F} \{\phi_k \Phi_m\}_{k \in \mathcal{J}_m} \right)$$

where the subset of indices $\mathfrak{M}_F \subset \mathfrak{M}$ corresponds to the traces on F , and Φ_m and \mathcal{J}_m are the enriching function and the set of enriched nodes relative to m -th trace.

We end briefly recalling how enriching functions are selected in the DFN context, referring the reader to [19] for more details in general cases and [1, 2] for details in the DFN simulations. For each fracture F , let S_m , $m \in \mathfrak{M}_F$ be a trace on F . We distinguish two cases: a) S_m is entirely crossing the fracture (the trace is hence a so called closed interface); b) one or more endpoints of S_m lie inside F (open interface). In the case of closed interfaces, the enriching function Φ_m is suitably set as $\Psi^m(\hat{x}) = \|\bar{x} - \hat{x}\|$, where \bar{x} is the projection of \hat{x} on S_m (see [17]).

In the case of open interfaces, Φ_m is still used for reproducing non-smooth behaviour on elements intersecting the trace but not containing trace tips. For each trace tip lying inside F , we also add new enriching functions (see [17]) defined as follows. Let $\sigma_m = \{s^1, s^2\}$ be the set of trace tips of S_m . If s^ℓ lies inside F , we introduce the enriching functions

$$\Theta_{s^\ell}^m(\mathbf{x}) \in \left\{ r \cos \frac{\theta}{2}, r^2 \cos \frac{\theta}{2}, \sqrt{r} \cos \frac{\theta}{2} \right\}, \quad s^\ell \in \sigma_m$$

where r is the distance between the current point and trace tip, and θ is the polar angle of \hat{x} with respect to a reference system centred into trace tip with the x -axis aligned to the trace, and oriented such that the trace lies

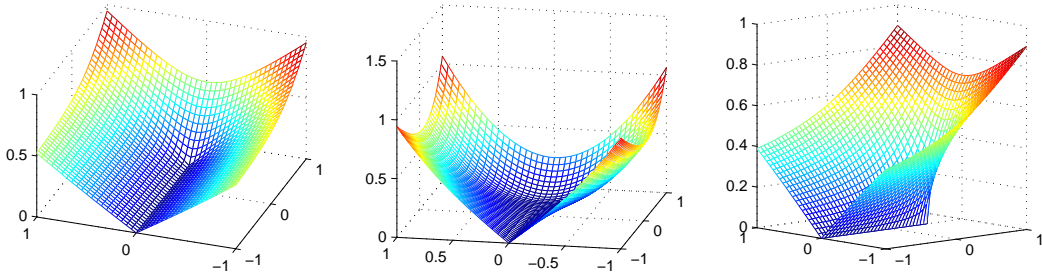


Figure 1: Behaviour of trace tip enrichment functions

on the negative side. Tip enrichments are introduced only on elements containing traces endpoints. Functions $\Theta_{s^\ell}^m(\mathbf{x})$ are plotted, from left to right, in Figure 1. We remark that the choice of enrichments is quite arbitrary. The selection here adopted is well suited to describe the nonsmooth behaviour of the solution around trace tip. Other choices are possible, as well as the use of a larger number of enrichments around the tip. This latter possibility could improve the description of the solution, but would increase the overall computational cost.

We refer the reader to [19, 1, 2] for more details about implementation of the XFEM, which include for example methods to preserve FEM optimal convergence rates and correctly perform accurate numerical integration of irregular functions.

5 Discretization of the constrained optimization problem

Following the paradigm “First optimize then discretize” we now describe the discrete version of the method introduced in the previous section.

Let us introduce an independent triangulation $\mathcal{T}_{\delta,i}$ on each fracture F_i , $\forall i \in \mathcal{J}$. Let $V_{\delta,i}$ be the finite dimensional trial and test spaces defined on the elements of $\mathcal{T}_{\delta,i}$ and spanned by Lagrangian basis functions $\phi_{i,k}$, $k \in \mathcal{I}_i = \{1, \dots, N_i\}$. Let us denote by h_i the discrete approximation of H_i , $i \in \mathcal{J}$:

$$h_i(\mathbf{x}) = \sum_{k=1}^{N_i} h_{i,k} \phi_{i,k}(\mathbf{x}), \quad \forall i \in \mathcal{J}.$$

The algebraic formulation of the operator A_i in equation (5) is the usual one:

$$(A_i)_{k\ell} = \int_{F_i} \nabla \phi_{i,k} \nabla \phi_{i,\ell} \, dF_i + \alpha \sum_{s \in \mathcal{S}_i} \int_S \phi_{i,k|_S} \phi_{i,\ell|_S} \, d\gamma,$$

where, overloading notation, we denote by $A_i \in \mathbb{R}^{N_i \times N_i}$, $i \in \mathcal{J}$, also the matrix defining the discrete algebraic operator. For all $S \in \mathcal{S}$, let us fix a finite dimensional subspace of \mathcal{U}^S for the discrete approximations u_i^S and u_j^S of the control variables U_i^S and U_j^S . In the discrete version of the problem we choose $\mathcal{U}^S = L^2(S)$ and thus we can remove the constraint of disjoint traces made in Section 3 (see [1]). Let $\{\psi_k^S\}_{k=1, \dots, N_S}$ be the basis introduced on the discrete control space on trace S . For application of gradient based methods, we choose a common arbitrary basis for u_i^S and u_j^S , $i, j \in I_S$, not necessarily depending neither on the mesh on F_i , nor on

the mesh on F_j . So we write

$$u_l^S = \sum_{k=1}^{N_S} u_{l,k}^S \psi_k^S \quad \forall l \in I_S, S \in \mathcal{S}_i.$$

For each fracture F_i , we set $N_{\mathcal{S}_i} = \sum_{S \in \mathcal{S}_i} N_S$ as the number of DOFs on traces of F_i . Given an index $i \in \mathcal{I}$ and a fracture $S \in \mathcal{S}_i$, we define matrices $B_i^S \in \mathbb{R}^{N_i \times N_S}$ as

$$(B_i^S)_{k\ell} = \int_S \phi_{i,k|_S} \psi_l^S d\gamma.$$

Matrices B_i^S , $S \in \mathcal{S}_i$, are then grouped row-wise to form the matrix B_i , which acts on a column vector u_i containing all the $N_{\mathcal{S}_i}$ control DOFs corresponding to the traces of F_i , obtained collecting vectors u_i^S , for $S \in \mathcal{S}_i$, with the same ordering introduced for the traces on F_i and used in the definition of B_i . For each fracture F_i let us introduce vectors $h_i \in \mathbb{R}^{N_i}$, $h_i = (h_{i,1}, \dots, h_{i,N_i})^T$, and $p_i \in \mathbb{R}^{N_i}$, $p_i = (p_{i,1}, \dots, p_{i,N_i})^T$. Furthermore, we define vectors $u \in \mathbb{R}^{N^T}$, with $N^T = \sum_{i \in \mathcal{I}} N_{\mathcal{S}_i}$, and $h \in \mathbb{R}^{N^F}$, with $N^F = \sum_{i \in \mathcal{I}} N_i$, as $u = (u_1^T, \dots, u_{\#\mathcal{I}}^T)^T$ and $h = (h_1^T, \dots, h_{\#\mathcal{I}}^T)^T$. The algebraic formulation of the primal equations (5) is then

$$A_i h_i = \tilde{q}_i + B_i u_i, \quad i \in \mathcal{I}, \quad (15)$$

where \tilde{q}_i accounts for the term q_i in (5) and for the weak discrete imposition of boundary conditions on the fracture F_i . We proceed similarly for the equations (10), (11) and (13), in which the operators C_i^S and B_i^* , $i \in \mathcal{I}$, are nothing but restriction operators. We thus obtain the algebraic equations for the definition of the discrete approximations p_i and δh_i . Further, given two indices $q, r \in \mathcal{I}$ (possibly $q = r$), we define matrices $C_{q,r}^S$ and $C_{q,r}$ as

$$(C_{q,r}^S)_{k\ell} = \int_S \phi_{q,k|_S} \phi_{r,\ell|_S} d\gamma, \quad C_{q,r} = \sum_{S \in \mathcal{S}_q} C_{q,r}^S.$$

The discrete counterpart of equations (10) and (13) $\forall i \in \mathcal{I}$ are

$$A_i p_i = C_{i,i} h_i - \sum_{j \in J_i} C_{i,j} h_j - \alpha [B_i u_i + \sum_{j \in J_i} B_j u_j - \alpha (C_{i,i} h_i + \sum_{j \in J_i} C_{i,j} h_j)], \quad (16)$$

$$A_i \delta h_i = B_i \delta u_i. \quad (17)$$

The discrete gradient of the functional $J(U)$ and the optimal step-size λ become

$$\nabla J(u_i) = P_{i|_{\mathcal{S}_i}} + u_i - \alpha h_i(u_i)_{|\mathcal{S}_i} + \sum_{j \in J_i} (u_j|_{\mathcal{S}_i} - \alpha h_j(u_j)_{|\mathcal{S}_i}), \quad (18)$$

$$\nabla J(u) = \prod_{i \in \mathcal{I}} \nabla J(u_i), \quad (19)$$

$$\lambda = - \sum_{i \in \mathcal{I}} (\nabla J(u_i), \delta u_i)_{\mathcal{S}_i} \left\{ -2\alpha \sum_{i \in \mathcal{I}} (\delta u_i + \delta u_j|_{\mathcal{S}_i}, \delta h_i|_{\mathcal{S}_i})_{\mathcal{S}_i} + \sum_{i \in \mathcal{I}} \left(\|\delta h_i|_{\mathcal{S}_i} - \delta h_j|_{\mathcal{S}_i}\|_{\mathcal{S}_i}^2 + \|\delta u_i + \delta u_j|_{\mathcal{S}_i}\|_{\mathcal{S}_i}^2 + \alpha^2 \|\delta h_i|_{\mathcal{S}_i} + \delta h_j|_{\mathcal{S}_i}\|_{\mathcal{S}_i}^2 \right) \right\}^{-1}. \quad (20)$$

We end this Section with the discrete version of Algorithm 3.

Algorithm 5.

1. Set $k = 0$ and initial guess for control variable u^0 ;
2. find $h^0 = h(u^0)$ solving on each fracture (15);
3. find $p(u^0)$ solving on each fracture (16);
4. evaluate $\nabla J(u^0)$ by (19);
5. set $(\delta u)^0 = -\nabla J(u^0)$;
6. While(stopping criterion not satisfied)
 - 6.1. compute optimal step-size λ^k along direction $(\delta u)^k$ by (20);
 - 6.2. set $u^{k+1} = u^k + \lambda^k (\delta u)^k$;
 - 6.3. $\forall i \in \mathcal{I}$ find $h_i(u^{k+1})$ by (15);
 - 6.4. $\forall i \in \mathcal{I}$ find $p_i(u^{k+1})$ by (16);
 - 6.5. evaluate $\nabla J(u^{k+1})$;
 - 6.6. set $\beta^{k+1} = \|\nabla J(u^{k+1})\|_S^2 / \|\nabla J(u^k)\|_S^2$
 - 6.7. set $(\delta u)^{k+1} = -\nabla J(u^{k+1}) + \beta^{k+1} \delta u^k$
 - 6.8. $k = k + 1$;

We notice that, thanks to the minimization approach adopted, only the solution of many simple problems on the fractures is required, with a small exchange of trace-related data among them. This algorithm is therefore suitable for massively parallel computers and GPU-based computers.

5.1 Stopping criterion

The stopping criterion used in Algorithm 5 plays a relevant role for efficiency reasons. In fact, due to the arbitrary intersections of the traces with elements independently placed on each fracture, the two terms of the functional J do not vanish, in general. This follows from the fact that on each trace S the discrete functions $h_{i|_S}$ and $h_{j|_S}$ with $i, j \in I_S$ are piecewise polynomials on different partitions of the trace. As a consequence, $\delta h_{i|_S} - \delta h_{j|_S}$ is typically different from zero. Appropriate choice for stopping criteria is crucial in order to prevent a premature stop of the algorithm far from the minimum of the functional, avoiding at the same time useless iterations when we are already close to the minimum, when only negligible further reduction of the functional could be achieved at the expenses of a large number of additional iterations. We will discuss this in the next Section.

6 Numerical Results

In this section we present some numerical experiments aiming at showing the behaviour of our algorithm in relation to various stopping criteria and the quality of the solution obtained. After introducing the DFN problems used for the simulations, and discussing stopping criteria used in our computations, we analyze the solution obtained on the most complex DFN configuration investigated.

6.1 DFN configurations

A set of four different DFN configurations is considered with an increasing number of fractures and traces as described in Table 1.

Table 1: Problems description

Label	# \mathcal{J}	# \mathcal{M}	DOFs (coarse grid)		DOFs (fine grid)	
			h	u	h	u
7fract	7	11	1140	163	4007	378
11fract	11	26	2244	337	7172	825
50fract	50	153	13211	2187	42161	5166
100fract	100	313	26512	4637	85900	10978

In Figure 2 we show section on the $x - y$ plane of fracture systems. All fractures extend, orthogonally to $x - y$ plane, from $z = 0$ to $z = 1$, except for fractures in dashed line that range between $z = 0$ and $z = 0.5$. Homogeneous or non-homogeneous Dirichlet boundary conditions are prescribed on the sides marked with a dark circle or with a dark rectangle respectively, while homogeneous Neumann conditions are set on the other edges. Problem formulation is as in equation (1), where the transmissivity is assumed constant and equal to 1, and the source term is $q = 0$ on all the fractures. For the discretization we use first order Lagrangian finite elements and two different grids: a coarse grid with about 35 elements per unit area and a finer grid with about 100 elements per unit area. The corresponding number of DOFs is reported in Table 1. In all cases we set the parameter $\alpha = 0.5$ in the definition of the control variable and the starting guess for the control variable is $u^0 = 0$. For each configuration and grid, we define a reference solution obtained performing a large number of gradient iterations in order to safely approach the minimum of the functional. As an example, to highlight the complete non conformity of the mesh to the traces, we show in Figure 3 a zoom of the coarse grid for the DFN problem with eleven fractures.

6.2 Stopping criteria

For each problem and grid a large set of simulations is performed, considering the different stopping criteria described in the following.

In Figure 4 and Figure 5 we plot, for the various problems considered and for increasing number of iterations, scaled by the number of problem traces, the distance in H^1 -norm between the reference solution and the current solution, relative to the H^1 -norm of the reference solution: $\|h_{curr} - h_{ref}\|_{H^1} / \|h_{ref}\|_{H^1}$. The reference solution

Table 2: Exit criteria used in simulations

Label	Criterion
t_1	$\mathcal{R}_1 = J^k - J^{k-1} < tol_1$
t_2	$\mathcal{R}_2 = \ u^k - u^{k-1}\ < tol_2$
t_3	$\mathcal{R}_3 = J^k(J^k - J^{k-1}) < tol_3$

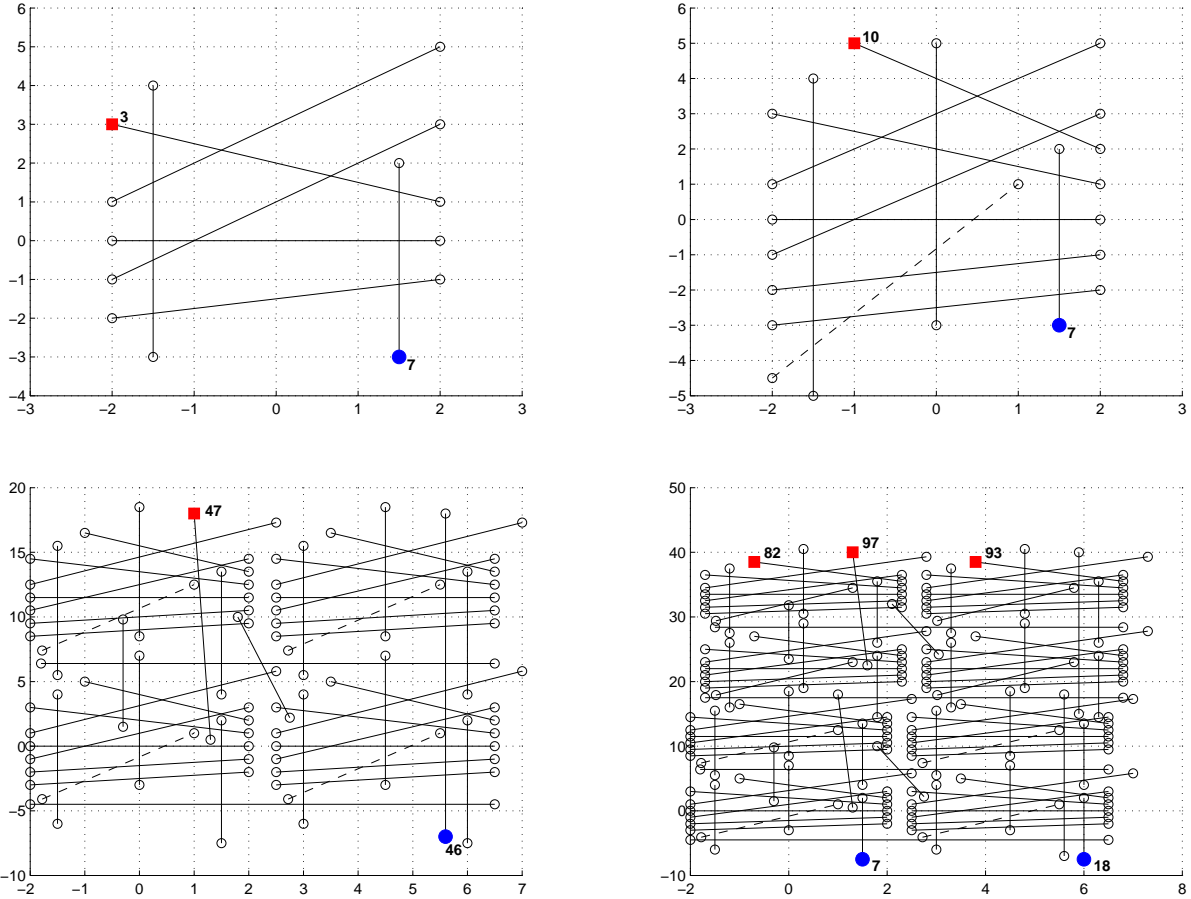


Figure 2: DFN configurations, section on $x - y$ plane. Left to right, top to bottom: **7fract**, **11fract**, **50fract**, **100fract**. Number is reported for fractures carrying Dirichlet boundary conditions (squared edge non homogeneous, filled circle homogeneous).

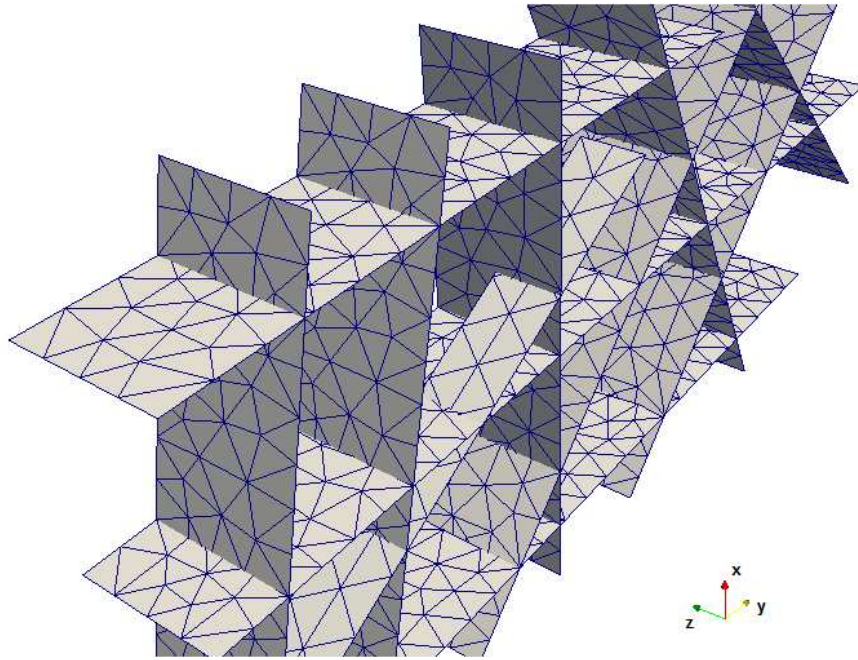


Figure 3: Zoom of grid for **11fract** problem.

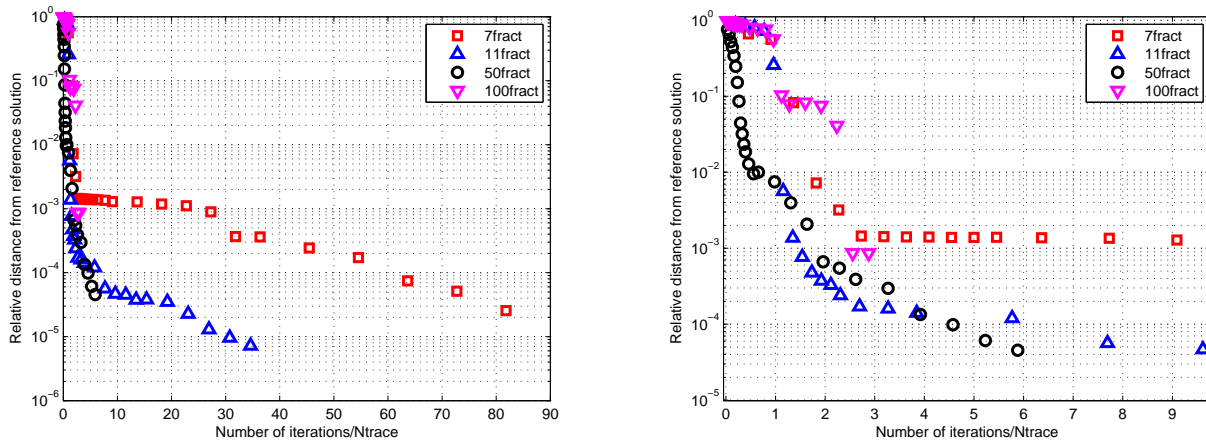


Figure 4: Relative distance in H^1 -norm of solution at different number of iterations, coarse grid. Right: zoom at lower number of iterations.

is obtained on the same grid, performing a very large number of conjugate gradient iterations. Figure 4, on the left, gives an overview on a wide range of iterations for the coarse grid, while on the right provides a zoom at lower iterations. Figure 5 provides a similar zoom for the finer grid. It should be noticed that the curves show initially a strong decreasing trend and, after a number of iterations that is few times the number of problem traces, variations of the solution with respect to the reference solution become smaller than 1%. Afterwards, the curves become almost flat and a large number of iterations would be required for negligible improvements in the solution. Therefore, we can see that the algorithm can provide a solution close to the best solution achievable within a reasonably small number of iterations, this number being proportional to the total number of traces in the system, with a proportionality factor in the order of few units.

As mentioned in Subsection 5.1, functional minimum is an unknown value different from zero. Hence, the choice of an exit criterion able to stop iterations when we are close enough to the solution, while avoiding useless iterations, is a crucial point. In Table 2 we report three possible criteria. Condition t_1 detects small variations in the functional values. Since the functional descent path can be step-like (see Figure 6 for an example), in order to avoid premature stops, the algorithm is terminated when $\mathcal{R}_1 < tol_1$ for a fixed number of subsequent iterations (six, in our computations). Approaching functional minimum we have that $\mathcal{R}_1 \rightarrow 0$. In Figure 7, left, we show the relative distance of the computed solution from the reference solution, corresponding to several values of the tolerance tol_1 . It can be noticed that a value around $tol_1 = 10^{-6}$ provides a good solution for all the problems considered.

Similarly, condition t_2 seeks small variations in the control variable. Again, to take into account possible temporary stagnation during the descent process, iterations are stopped when $\mathcal{R}_2 < tol_2$ six times subsequently. Also in this case as the functional approaches its minimum \mathcal{R}_2 tends to zero. We can see in Figure 7, middle, the behaviour of the solution in relation to the choice of tol_2 . The value $tol_2 = 10^{-7}$ appears to be a good choice.

As a possible alternative, criterion t_3 aims at detecting functional minimum, again avoiding premature stop at large values of the functional due to local stagnation. The rationale behind this criterion is to avoid stopping the iterates when $J^k - J^{k-1}$ is small but J^k is not small as well. Algorithm is then stopped the first time that $\mathcal{R}_3 < tol_3$. Also in this case \mathcal{R}_3 can be arbitrarily reduced with iterations. We plot solution behaviour in relation to tol_3 in Figure 7, right. We notice that in this case low tolerance values, around $tol_3 = 10^{-8}$, should be chosen.

6.3 DFN system solution

We now show the quality of the numerical solution obtained on the more complex DFN configuration considered herein. First we show in details the results obtained on two of the fractures in the **100fract** system: the source fracture 82 and the sink fracture 18 (see Figure 2). On the coarse grid, in Figures 8 and 9, left, we compare the solution on fracture traces, $\{h_{i|S}\}_{S \in \mathcal{S}_i}$, $i = \{18, 82\}$, and the solution on the traces of intersecting fractures, $\{h_j\}$ with $j \in J_i$. We can see a very good agreement between these values, ensuring the global continuity of the hydraulic head. In the right part of the same figures, we compare the co-normal derivative of solution on the traces of the current fracture and on trace-twin fracture (with opposite sign). In these figures $\phi(h) = \left[\left[\frac{\partial h}{\partial \nu_S} \right] \right]_S$. Again, we can observe, as expected, a very good agreement between these values, ensuring flux conservation.

In Figure 10 we show, for the same fractures, the solution on the traces obtained with four different meshes. Reported results show that, under grid refinement, the computed solutions are clearly approaching the same values. In Figure 11 we plot the whole solution obtained with the coarse grid on the fractures 82 and 18. In Figures 12 and 13 we report 3D pictures representing the DFN. The computing meshes are drawn and the solution is reported on the fractures using a colorbar. The arrows point the source fracture 93 and the sink fracture 7.

In Figure 14, left, the L^2 -norm of solution against iterations is plotted. The table of Figure 14, right, gives

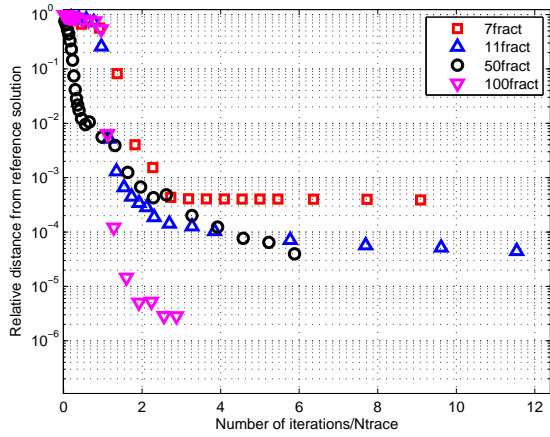


Figure 5: Relative distance in H^1 -norm of solution at different number of iterations, finer grid. Zooming at lower number of iterations.

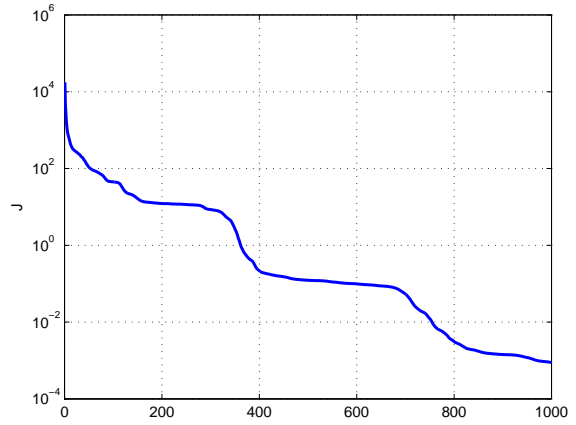


Figure 6: Example of functional step-like descent path. Problem **100fract** on the coarse grid.

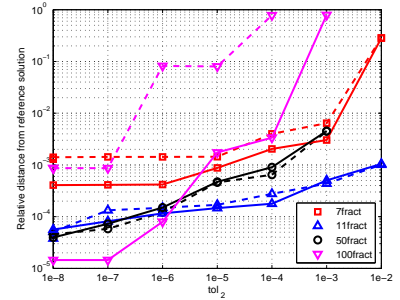
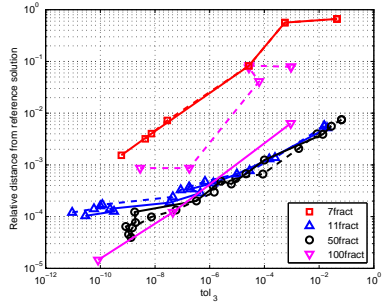
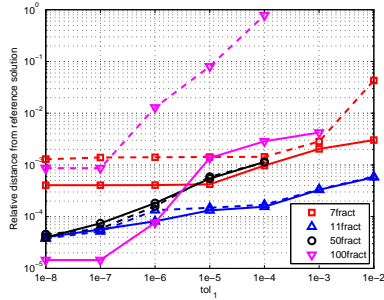


Figure 7: Relative distance in H^1 -norm from reference solution for different tolerances and stopping criteria. Left: condition t_1 ; middle: condition t_2 ; right: condition t_3 . Coarse grid in dashed line, finer grid in solid line.

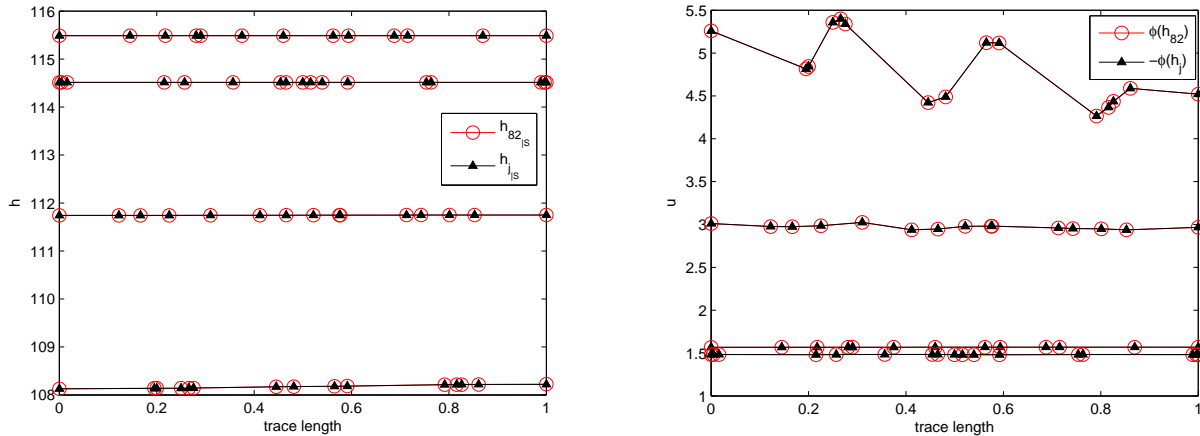


Figure 8: Problem **100fract**, source fracture 82, coarse grid. Solution on the traces (left) and co-normal derivative (right) compared with corresponding values computed on trace-twin fractures.

an indication of the conservativity of the method on the whole network of fractures, as it reports the values of the total fluxes discharged by the source fractures to the system and the total flux received by the sink fractures from the system. As expected the data match very well.

7 Conclusions

Major drawbacks in DFN numerical simulations lie in the definition of a good quality finite element triangulation and in the huge computational demand. The method introduced in [1, 2] and further developed in the present work provides a possible approach for circumventing these difficulties. The proposed method allows a fully independent triangulation on each fracture, thus eliminating any mesh related problem. Further, the method can be easily implemented on parallel machines, since the DFN simulation is split in many sub-problems on each fracture that can be solved independently by parallel processes, with little exchange of trace related data between trace-twin processes.

The contribution of the present work to the method is twofold. We introduce a new definition of the control variable for the optimal problem in order to eliminate the requirement of having a non-empty portion of the boundary of each fracture with Dirichlet boundary condition. We also introduce a conjugate gradient method for the optimization process in order to speed up convergence. By means of several numerical results we show that our algorithm provides a good quality solution within a small number of iterations that increases linearly with the number of traces in the system. The proportionality factor is in the order of few units for all the problems considered. The main computational effort in each iteration is devoted to the resolution of the linear systems on the fractures, that however are independent each other. Assuming that these linear systems have a comparable dimension, the total cost of each iteration scales as the number of fractures. Effectiveness of some stopping criteria for the gradient iterations is also discussed.

Further developments on the topic should include on one side an investigation of the scalability of the algorithm using an increasing number of parallel processes on different parallel architectures, and on another side the analysis of non-stationary problems. In the non-steady case, in order to reduce the computational effort,

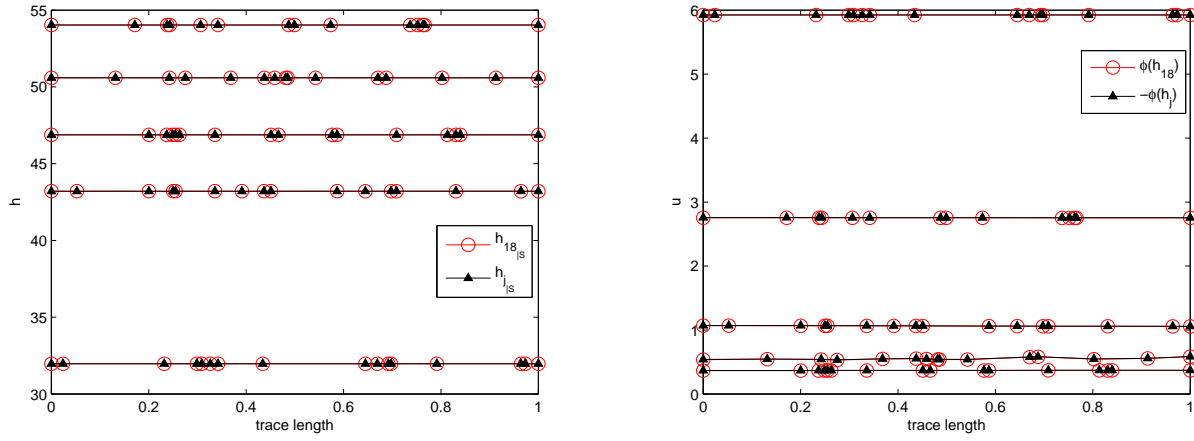


Figure 9: Problem **100fract**, sink fracture 18, coarse grid. Solution on the traces (left) and co-normal derivative (right) compared with corresponding values computed on trace-twin fractures.

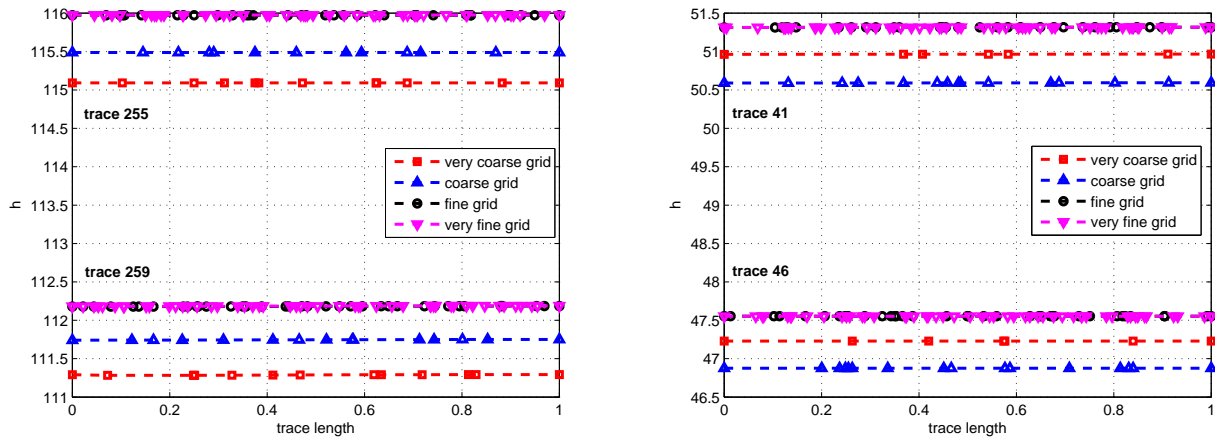


Figure 10: Solution on the traces of source fracture 82 (left) and sink fracture 18 (right) for two different grids, **100fract** problem.

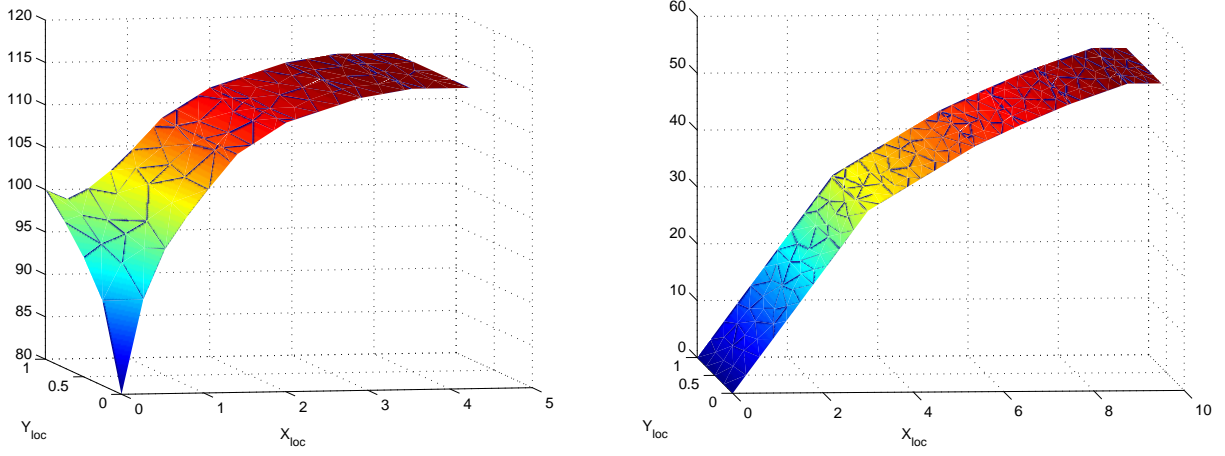


Figure 11: Solution on source fracture 82 (left) and on sink fracture 18 (right) for **100fract** problem. Reference system is local. Coarse grid.

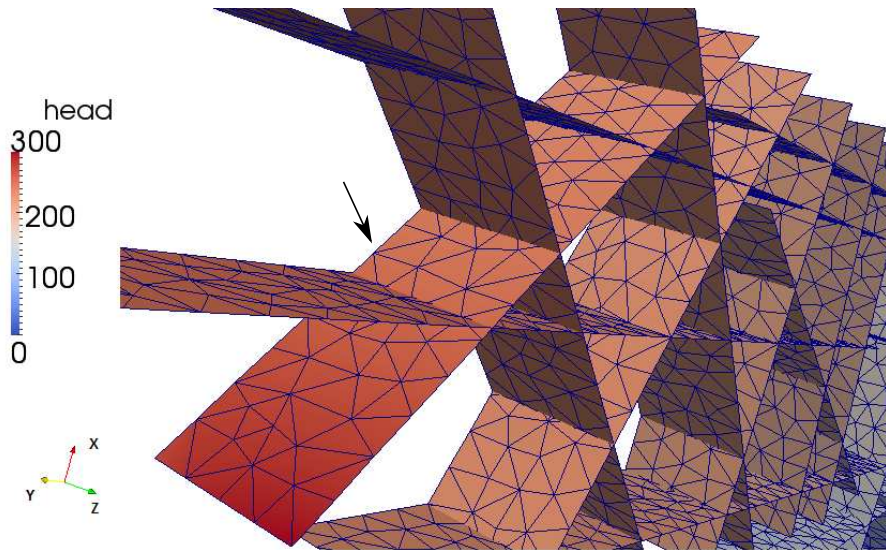


Figure 12: Solution on the DFN **100fract**. Arrow points source fracture 93.

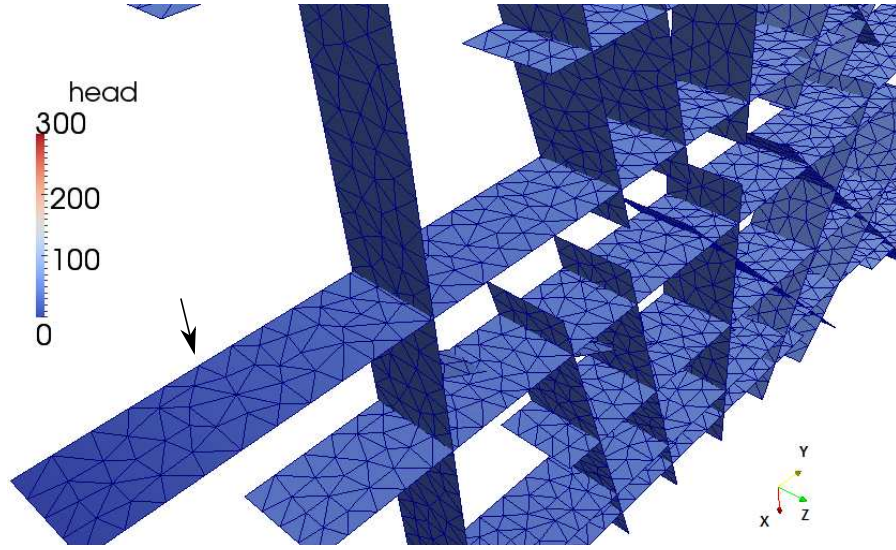
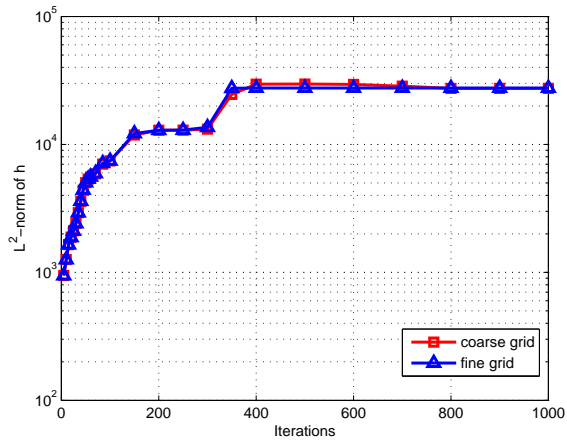


Figure 13: Solution on the DFN **100fract**. Arrow points sink fracture 7.



Grid	Source	Sink
very coarse grid	-15.12350	15.11294
coarse grid	-15.34119	14.99621
fine grid	-15.21692	15.21691
very fine grid	-15.21841	15.21653

Figure 14: Problem **100fract**: L^2 -norm of solution against iterations (left) and total fluxes on the source and sink fractures (right).

the application of reliable and efficient space-time a posteriori error estimates and adaptive algorithms, following the approaches of [20, 21], could be fruitful.

Acknowledgments

This work was financed by Italian funds from MIUR (PRIN grant) and INdAM-GNCS funds.

References

- [1] S. Berrone, S. Pieraccini, S. Scialó, A PDE-constrained optimization formulation for discrete fracture network flows, *SIAM Journal on Scientific Computing* To appear. Preprint available at <http://porto.polito.it/id/eprint/2495599>.
- [2] S. Berrone, S. Pieraccini, S. Scialó, On simulations of discrete fracture network flows with an optimization-based extended finite element method, *SIAM Journal on Scientific Computing* To appear. Preprint available at <http://porto.polito.it/id/eprint/2499005>.
- [3] W. S. Dershowitz, H. H. Einstein, Characterizing rock joint geometry with joint system models, *Rock Mechanics and Rock Engineering* 1 (1988) 21–51.
- [4] M. Vohralík, J. Maryška, O. Severýn, Mixed and nonconforming finite element methods on a system of polygons, *Applied Numerical Mathematics* 51 (2007) 176–193.
- [5] 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.
- [6] J. Erhel, J.-R. de Dreuzy, B. Poirriez, Flow simulation in three-dimensional discrete fracture networks, *SIAM J. Sci. Comput.* 31 (4) (2009) 2688–2705.
- [7] B. Noëtinger, 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.
- [8] G. Pichot, J. Erhel, J.-R. De Dreuzy, A mixed hybrid Mortar method for solving flow in discrete fracture networks, *Applicable Analysis* 89 (10) (2010) 1629–1643.
- [9] G. Pichot, J. Erhel, J.-R. 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.
- [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.
- [11] A. W. Nordqvist, Y. W. Tsang, C. F. Tsang, B. Dverstop, J. Andersson, A variable aperture fracture network model for flow and transport in fractured rocks, *Water Resource Res.* 28 (1992) 1703–1713.
- [12] 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.
- [13] J. Nocedal, S. Wright, *Numerical Optimization*, Springer, 1999.

- [14] T. Belytschko, T. Black, Elastic crack growth in finite elements with minimal remeshing, *Internat. J. Numer. Methods Engrg.* 45 (1999) 601–620.
- [15] N. Möes, J. Dolbow, T. Belytschko, A finite element method for crack growth without remeshing, *Internat. J. Numer. Methods Engrg.* 46 (1999) 131–150.
- [16] C. Daux, N. Möes, J. Dolbow, N. Sukumar, T. Belytschko, Arbitrary branched and intersecting cracks with the extended finite element method, *Internat. J. Numer. Methods Engrg.* 48 (2000) 1741–1760.
- [17] T. Belytschko, N. Möes, S. Usui, C. Parimi, Arbitrary discontinuities in finite elements, *Internat. J. Numer. Methods Engrg.* 50 (2001) 993–1013.
- [18] I. Babuška, J. M. Melenk, The partition of unity method, *Internat. J. Numer. Methods Engrg.* 40 (4) (1997) 727–758.
- [19] T.-P. Fries, T. Belytschko, The extended/generalized finite element method: an overview of the method and its applications, *Internat. J. Numer. Methods Engrg.* 84 (3) (2010) 253–304.
- [20] S. Berrone, Robust a posteriori error estimates for finite element discretization of the heat equation with discontinuous coefficients, *M2AN* 40 (6) (2006) 991–1021.
- [21] S. Berrone, A local-in-space-timestep approach to a finite element discretization of the heat equation with a posteriori estimates, *SIAM J. Numer. Anal.* 47 (4) (2009) 3109–3138.