# Modeling of transport processes through large-scale discrete fracture networks using conforming meshes and open-source software

Tri Dat Ngo*, André Fourno, Benoit Noetinger

*IFP Energies Nouvelles, 1&4 Avenue de Bois Préau, 92500 Rueil-Malmaison, France*

**Abstract**

Most industrial and field studies of transport processes in Discrete Fracture Networks (DFNs) involve strong simplifying assumptions, especially at the meshing stage. High-accuracy simulations are therefore required for validating these simplified models and their domain of validity. The present paper proposes an efficient workflow based on open-source software to obtain transport simulations. High-quality computational meshes for DFNs are first generated using the conforming meshing approach FraC. Then, a tracer transport model implemented in the open-source code DuMux is used for simulating tracer transport driven by the advection-dispersion equation. We adopt the box method, a vertex-centered finite volume scheme for spatial discretization, which ensures concentration continuity and mass conservation at intersections between fractures. Numerical results on simple networks for validation purposes and on complex realistic DFNs are presented. An a-posteriori convergence study of the discretization method shows an order of convergence $O(h)$ for tracer concentration with $h$ the mesh size.

*Keywords:* DFN, Conforming mesh, Transport processes, Vertex-centered finite volume method

## 1. Introduction

During recent decades, modeling of transport processes in fractured porous media has received increasing attention from the geoscientist community because of its wide range of applications in geothermal energy (Aquilina et al., 1998), petroleum exploration and production (Bourbiaux, 2010), $CO_2$ geological storage and nuclear waste disposal (Grenier et al., 2005; Fourno et al., 2007). Fracture networks might consist of pre-existing natural or pressure-induced fractures in the case of hydraulic fracturing (Delorme et al., 2016). Solute transport in fractured media should be studied using either continuous (Svensson, 2001; Karimi-Fard et al., 2006; Fourno et al., 2013) or discrete models (Erhel et al., 2009; Hyman et al., 2014; Berrone et al., 2017). When fractures are densely distributed and the length of fractures is of the same order of magnitude, continuous models can be used safely, because there is a representative elementary volume (REV) (Long et al., 1982; Neuman, 1988). At scales larger than the REV, the contribution of fractures may be captured in equivalent parameters that are obtained via upscaling processes (Karimi-Fard et al., 2006; Matthai and

*Corresponding author
*Email address:* tri-dat.ngo@ifpen.fr (Tri Dat Ngo)

Nick, 2009; Jourdain et al., 2014). On the other hand, for sparsely distributed fracture networks, possibly no REV exists; the solute is carried along connected preferential pathways of high permeability. In that situation, in order to capture the effects related to the fracture sparseness, the domain should be modeled by discrete approaches. The basic idea is to keep an explicit representation of fracture networks. In this work, we focus on solute transport in discrete fracture networks (DFNs) assuming that flow only occurs through fracture planes, the matrix surrounding fractures is disregarded.

Fractures are generally considered as lower-dimensional objects, i.e. 1D or 2D objects embedded in a 3D domain. Pipe-network approaches, which are based on the assumption of flow channeling within DFNs, model fractures as 1D-in-3D objects. With this strong topological simplification, pipe-network models allow to significantly reduce the computational cost for large-scale reservoirs. The accuracy of such models has been investigated for steady-state flow and transport (Cacas et al., 1990; Dershowitz and Fidelibus, 1999; Gylling et al., 1999). Noetinger and Jarrige (2012) recently proposed a quasi steady-state method for solving transient Darcy flow in complex 3D fractured networks only. This work is then extended for accounting for transfers arising from the matrix (Noetinger, 2015). At the lowest order of approximation, this method needs only one unknown that is assigned at each identified intersection location between fractures. The pipe network model is then recovered. Nevertheless, further studies should focus on validation of the pipe-models for transient flow and transport.

Another simplifiying assumption is to consider fractures as planar objects. High-accuracy simulations of flow and transport within DFNs can be obtained on conforming or non-conforming meshes. By using non-conforming meshes, advanced numerical approaches, such as the PDE-constrained optimization approach (Berrone et al., 2013; Benedetto et al., 2014, 2016; Berrone et al., 2017) and the Mortar method (Erhel et al., 2009; Pichot et al., 2010, 2012), allow to alleviate the meshing-related difficulties. Nevertheless, applying such a method may lead to a high additional computational effort and requires adaptations of classical numerical schemes to non-conforming meshes. Therefore, although creating high-quality conforming meshes for such networks is a non-trivial task due to geometrical complexities, it is still being dealt with by many recent studies (Hyman et al., 2014, 2015; Huang et al., 2016; Fourno et al., 2016).

Hyman et al. (2014) developed the feature rejection algorithm for meshing (FRAM) to generate three-dimensional conforming grids for DFNs. This approach is based on constraining generations of DFNs to exclude arbitrarily small features from the network, only features greater than a user-prescribed minimum length scale $h$ will be generated. Nonetheless, by removing small fractures accuracy of simulations can be affected due to the loss of fracture connectivity within the resulting mesh. Huang et al. (2016) and Fourno et al. (2016) recently proposed a simple yet efficient meshing approach for DFNs, the so-called "Fracture Cut Method for Meshing" (FraC). The main idea of the approach is to decompose each fracture into a set of connected closed contours, between which common segments will be discretized in a conforming manner. Even tiny fractures are taken into account to conserve the DFN connectivity, and a strategy for moving or adding intersecting points is applied to ensure an acceptable quality of the final mesh.

Furthermore, the resulting conforming meshes obtained from these meshing approaches have often a complex "non-manifold" topology (Sander et al., 2015). Handling such a situation usually requires the introduction of additional data fields and logic with respect to standard manifold grids. Some finite element codes using Mixed and Hybrid Finite Element Methods (MHFEM), for instance Cast3M (2017) (Fourno et al., 2016), allow to avoid this extra implementation effort. On the other hand, standard finite volume codes often only allow for manifold topologies that are predominant in realistic applications and do not address non-manifold issues. Currently, there are few finite volume simulators for subsurface flow and transport, e.g. PFLOTRAN (Lichtner et al., 2013; Hammond et al., 2014) and DuMux (Flemisch et al., 2011), able to deal with non-manifold topologies. Resulting meshes from the FRAM method were used to obtain single-phase flow solutions using PFLOTRAN, followed by Lagrangian-based particle transport modeling (Hyman et al., 2014; Makedonska et al., 2015; Karra et al., 2015; Hyman et al., 2015). The objective of this paper is to describe an efficient workflow that provides high-accuracy solute transport simulations using conforming meshes. The proposed workflow uses the FraC approach for meshing and DuMux for simulation frameworks in order to profit from its Euclidean-based transport modeling capability. It is noteworthy that other advanced conformal mesh generators such as dfnGen (Hyman et al., 2014) could be employed for the network generation and meshing, the FraC approach was chosen simply because of its user-friendliness. The workflow is able to handle with non-trivial DFN configurations, which remain a major challenge for classical methods. Furthermore, the numerical framework can easily be extended to consider multiphase, multi-component flow, transient single-phase flow or reactive transport as well as other hydrological modelling issues using DFN models.

The contribution is organized as follows. Section 2 provides the mathematical formulation and numerical methods for transport processes using the DuMux (Flemisch et al., 2011) code. Section 3 is focused on the methodology of the FraC approach used to obtain conforming meshes for DFNs. Numerical results of solute transport on simple and complex DFNs are shown in Section 4, both for model validation and demonstration purposes. For validation purpose, solutions of solute transport obtained with DuMux on a simple single-fracture DFN are compared to analytical solutions. Another validation on a more complex 7-fracture DFN is also proposed using numerical results from Ahmed et al. (2015) as reference. The second part of the section is dedicated to case studies of transport processes on a synthetic benchmark 33-fracture network and on a complex and realistic DFN in order to assess the reliability of the approach. Finally, we conclude the paper in Section 5 by giving some directions for further studies.

## 2. Mathematical formulation and numerical methods

The motion of a non-reactive solute in the subsurface can be described by the advection-diffusion equation, (Bear et al., 2012)as follows:

$$\omega \frac{\partial C}{\partial t} = \nabla.(\overline{\overline{D}}.\nabla C - C\mathbf{v}) + q, \tag{1}$$

where $\omega$ (-), $\overline{\overline{D}}$ ($L^2/T$) are the porosity and the dispersion coefficient tensor for the solute in water, respectively. $C$ ($M/L^3$) stands here for the solute concentration, $\mathbf{v}$ ($L/T$) for the velocity field and $q$ ($M/(L^3.T)$) for the external sink and/or source term. The velocity field $\mathbf{v}$ is given by the Darcy's law:

$$\mathbf{v} = -\frac{\overline{\overline{K}}}{\mu}(\nabla p + \rho \mathbf{g}), \tag{2}$$

in which $p$ ($M/(L.T^2)$) denotes the water pressure, $\mathbf{g}$ the gravity acceleration vector, $\overline{\overline{K}}$ ($L^2$) the permeability tensor, $\mu$ ($M/(L.T)$) the dynamic viscosity of water and $\rho$ ($M/L^3$) is its density. The water flow is assumed to be in the steady-state, therefore Equation (1) can be recast as:

$$\omega \frac{\partial C}{\partial t} = \overline{\overline{D}} \nabla^2 . C - \mathbf{v} . \nabla C + q. \tag{3}$$

The one-dimensional form of Equation (3) reads as follows

$$\omega \frac{\partial C}{\partial t} = D_L \frac{\partial^2 C}{\partial x^2} - v . \frac{\partial C}{\partial x} + q. \tag{4}$$

Although several physical models available in DuMux allow to study a wide range of subsurface problems, spanning from single-phase single-component isothermal to three-phase three-component non-isothermal flow, non-reactive solute transport is however not implemented in the code. Fortunately, there is the `1p2c` model simulating single-phase flow of a compressible fluid consisting of two components where the fluid properties such as the density and the viscosity depend upon the pressure. The primary variables of the `1p2c` model are the pressure $p$ and the mole or mass fraction of dissolved components $x^\kappa$. Basing on this model, transport processes can be simulated in DuMux with a slight modification of the code, in which the pressure-dependency of the fluid properties is removed. The steady-state fluid flow are considered, i.e. $\mathbf{v}$ is constant, therefore, the continuity equation of the `1p2c` model becomes the advection-diffusion equation.

The non-manifold issue of the mesh topology is solved in DuMux (2017) via the DUNE grid manager DUNE::FoamGrid (Sander et al., 2015). A vertex-centered finite volume scheme, also known as the *box method* (BM) (see, for example, Huber and Helmig (2000)), is chosen for solving flow and transport in porous media. Concretely, the computational domain $\mathcal{T}_h$ is discretized by a finite element grid, the so-called "primary mesh", where $h$ is the mesh size. Since fractures are considered as surface objects, the unit cell could be either triangles or quadrilaterals. A "secondary mesh" which describes the finite volumes is constructed based on the finite element grid. A finite volume $\mathcal{B}_i$, also known as a *box*, is obtained by connecting the gravity center of element $i$ with associated edge midpoints (see Figure 1). Thus, each finite volume is associated with a node of the primary mesh. Furthermore, the model variables are defined for control-volumes but not finite elements, they are thus assigned to the grid nodes. Moreover, local physical properties of porous media, such as permeability and porosity, are specified element-wise, the corresponding values assigned to

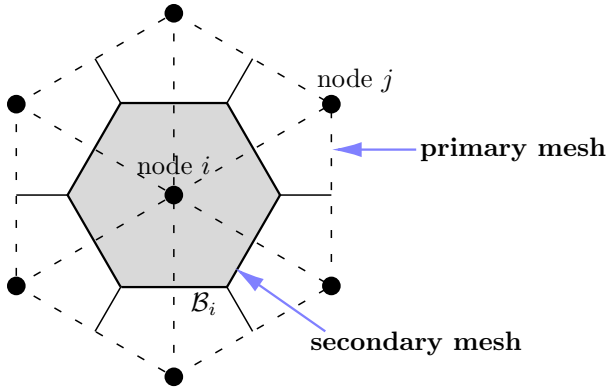



**Figure 1:** Spatial discretization of the box method including two meshes: "primary mesh" in dashed lines and "secondary mesh" in continuous lines. The gray polygon represents the box $\mathcal{B}_i$ corresponding to the node $i$.

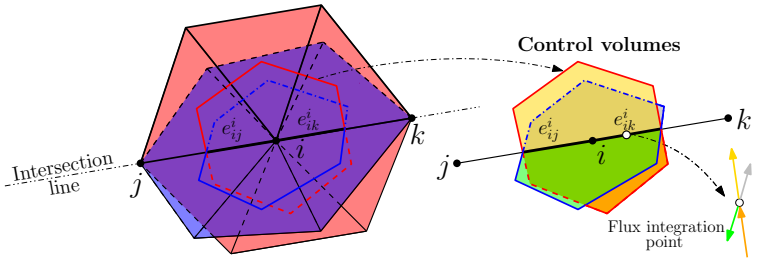



**Figure 2:** Illustration of the flux distribution for intersecting control volumes. The orange and yellow polygons belong to the red fracture, and the green and gray polygons belong to blue fracture. In this example, inflow occurs from the orange polygon and outflow occurs through the remaining polygons. This figure is reproduced from Hyman et al. (2015); Makedonska et al. (2015). We refer to the online version of the paper for the colors mentioned in this caption.

each control volume are determined by a weighted average of the properties of the sub-control volumes.

The BM inherits advantages from both the finite element and finite volume method: the finite element property makes it possible to deal with unstructured grids and the finite volume method ensures mass conservation. In addition, for flow and transport simulations on non-manifold conforming meshes, the BM unconditionally guarantees pressure continuity and mass conservation at the intersections between planar surfaces. Assume that we are considering two intersecting fractures (the red and blue ones) as in Figure 2. Since the mesh is conforming along the intersection line, the control volumes associated to the node i within each fracture share common edges ($e^i_{ij}$ and $e^i_{ij}$) of the sub-control volumes, on which the fluxes are computed. Then, mass conservation at the fracture intersection can naturally be ensured, and the box method is therefore chosen as discretization method for the following flow and transport simulations.

Moreover, although the temporal discretization may be done using an explicit or an implicit Euler scheme

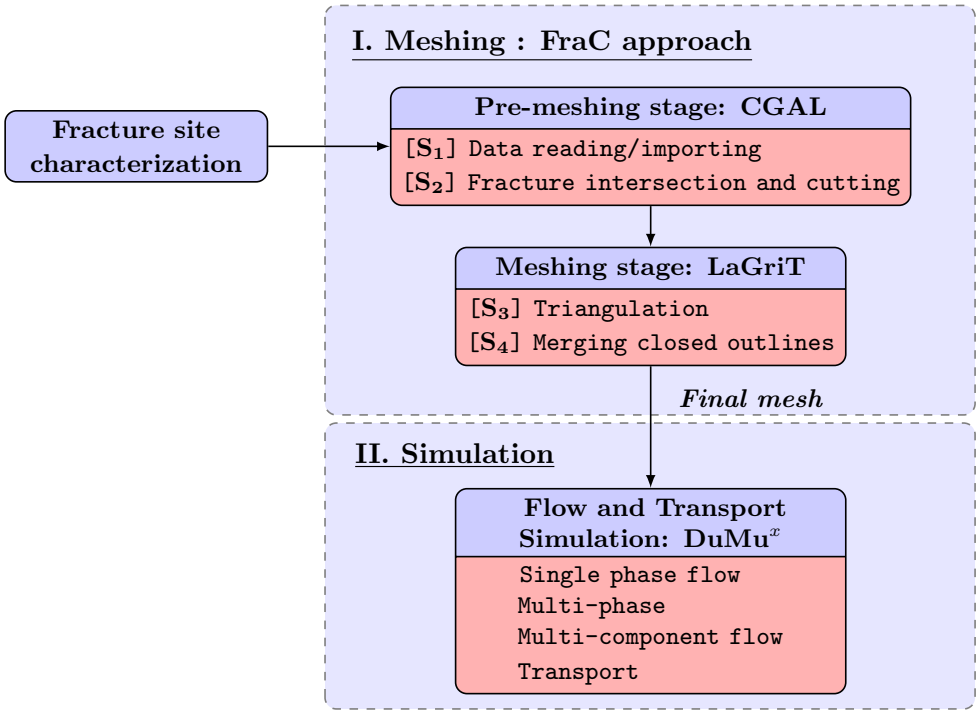



**Figure 3:** General flow chart from 3D mesh generation of DFNs basing on geological input datasets to numerical simulations of flow and transport processes. The framework consists of two major parts: (I) Mesh generation using the FraC approach and (II) Numerical simulations in DuMux. For the first part, the implementation of the FraC approach is carried out by using CGAL and LaGriT for pre-meshing and meshing stage, respectively. Detailed steps of each stage are also listed.

in DuMux, only implicit models are retained in our applications. The time-step size is controlled in an adaptive manner based on the number of iterations needed for the Newton's method to converge for the last time integration (Flemisch et al., 2011).

## 3. Mesh generation

### *3.1. Description of the meshing method*

The FraC method involves two primary stages, i.e. pre-meshing and meshing stages (see Figure 3. The pre-meshing stage includes two steps that are all implemented using the Computational Geometry Algorithms Library (CGAL): (i) data reading/ importing ($S_1$); (ii) fracture intersections and cutting ($S_2$). The triangulation ($S_3$) and merging closed contours ($S_4$) processes gather together in the meshing stage, which is done by using Los Alamos Grid Toolbox (2013) because CGAL is unfortunately not able to store non-manifold data. The main features in each stage can be summarized as follows.

*(i) $S_1$: Data reading/importing*

The input for FraC comes generally from fractured site characterizations that provide distributions of fracture forms, orientations, dimensions, and spatial locations. Fractures are then representatively approximated by concave polygons. Each polygon $\mathcal{F}_i$ will then be discretized with a user-defined characteristic length $h_i$, that is subsequently used as the target edge length for $\mathcal{F}_i$ in the meshing stage. It means that the edge length of the mesh triangles is of the same order of magnitude as $h_i$. Hence, resolution of the final mesh can be easily controlled by adjusting $h_i$; high values of $h_i$ result in coarse meshes and reducing $h_i$ yields mesh refinement. The characteristic length $h_i$ is fairly similar to the minimum length scale in the FRAM approach (Hyman et al., 2014, 2015).

*(ii) $S_2$: Fracture intersection and cutting*

The next step involves finding intersections between the polygons associated with the fractures. An octree-like approach (Khvoenkova and Delorme, 2011; Hyman et al., 2014) is used to optimize the localization of fractures in the space. More precisely, an axis-aligned minimum bounding box $BB_i$ is created around $\mathcal{F}_i$ where $BB_i$ is defined by the minimal and maximal value of the corresponding coordinate of the polygon vertices. Then a preliminary test is performed to check intersection between the bounding boxes $BB_i$ and $BB_j$ of two fractures $\mathcal{F}_i$ and $\mathcal{F}_j$ respectively. The number of intersection tests between bounding boxes performed for each fracture can be reduced using a three-dimensional octree structure based on fracture locations in the corridor. This approach works well if the size of the fractures is almost of the same order of magnitude.

If $BB_i$ and $BB_j$ are not touching in space, obviously $\mathcal{F}_i$ and $\mathcal{F}_j$ do not intersect. Otherwise, if $BB_i$ and $BB_j$ intersect, the intersection test between $\mathcal{F}_i$ and $\mathcal{F}_j$ is then performed. The intersection line $\mathcal{L}_{ij}$ between the $\mathcal{F}_i$ and $\mathcal{F}_j$ planes is first determined. Let $\partial\mathcal{F}_\alpha$ be the boundary of $\mathcal{F}_\alpha$ with $\alpha \in \{i, j\}$, then finding intersections between the fractures becomes finding intersections between $\partial\mathcal{F}_i$, $\partial\mathcal{F}_i$ with $\mathcal{L}_{ij}$. We denote $n_{ij}^{(\alpha)}$ the number of intersection points between $\partial\mathcal{F}_\alpha$ with $\mathcal{L}_{ij}$. So far, two cases can occur:

- $n_{ij}^{(i)} = 0$ or $n_{ij}^{(j)} = 0$: The fractures do not intersect.
- $n_{ij}^{(i)}, n_{ij}^{(j)} > 0$: In most cases, four intersection points $\mathbf{p}_i^{(1)}$, $\mathbf{p}_i^{(2)}$, $\mathbf{p}_j^{(1)}$, $\mathbf{p}_j^{(2)}$ ($n_{ij}^{(i)} = n_{ij}^{(j)} = 2$) may be found, where $\mathbf{p}_i^{(1,2)}$ lie on $\mathcal{F}_i$ and $\mathbf{p}_j^{(1,2)}$ lie on $\mathcal{F}_j$. Degenerated cases of tangent fractures where $n_{ij}^{(i)}$ or $n_{ij}^{(j)}$ or both of them equal to one may also exist. The intersection points are then sorted to find out endpoints. Let $\overline{\mathcal{I}}_{ij}$ be the segment connecting two endpoints, $\mathcal{I}_i$ be the portion of $\overline{\mathcal{I}}_{ij}$ which lies only on $\mathcal{F}_i$ but not on $\mathcal{F}_j$, $\mathcal{I}_j$ be the portion which lies only on $\mathcal{F}_j$ but not on $\mathcal{F}_i$, and $\mathcal{I}_{ij}$ be the portion which lies on both $\mathcal{F}_i$ and $\mathcal{F}_j$ (see Figure 4). Finally, the fractures intersect only if $\mathcal{I}_{ij} \neq \emptyset$.

The fracture cutting step expresses the key idea of the FraC approach, it is solely applied for intersecting fractures. Each of two intersecting fractures $\mathcal{F}_i$ and $\mathcal{F}_j$ will then be cut along the extended intersection $\overline{\mathcal{I}}_{ij}$ to obtain interconnected closed contours, subsequently referred to as "sub-fractures". To guarantee the

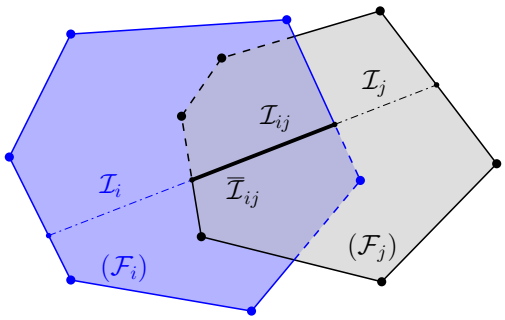



**Figure 4:** An example of two intersecting fractures $\mathcal{F}_i$ (blue) and $\mathcal{F}_j$ (gray). The extended intersection $\overline{\mathcal{I}}_{ij}$ (bold line) between them involves $\mathcal{I}_{ij}$ (bold line), $\mathcal{I}_i$ and $\mathcal{I}_j$ (dashed lines).

homogeneity of the final mesh, the sub-fractures and more precisely, the segment $\overline{\mathcal{I}}_{ij}$ should also be discretized according to the mesh characteristic length of the original fracture. For that, special focus is put on $\overline{\mathcal{I}}_{ij}$ which is also sub-divided in a homogeneous or adaptive manner. In the first case we define the target edge length $h_{ij} = min(h_i, h_j)$ common for $\overline{\mathcal{I}}_{ij}$ and its sub-segment including $\mathcal{I}_i$, $\mathcal{I}_j$, $\mathcal{I}_{ij}$. Here, $h_i$ and $h_j$ are the target edge length of $\mathcal{F}_i$ and $\mathcal{F}_j$ respectively. In the second case, only $\mathcal{I}_{ij}$ will be discretized with $h_{ij}$, the target edge length for $\mathcal{I}_i$ and $\mathcal{I}_j$ remains $h_i$ and $h_j$ respectively. At the endpoints of $\mathcal{I}_i, \mathcal{I}_j$ and $\mathcal{I}_{ij}$, two strategies are flexibly applied: depending on neighborhood area, discretization points of $\partial\mathcal{F}_\alpha$ are either moved (Type I) or created (Type II) to model the intersection points (Figure 5a). These strategies allow avoiding too short edges resulting in grid cells with extremely high aspect ratio in the final mesh.

Once all edges of sub-fractures are discretized, new intersection tests between them and other fractures/ sub-fractures are carried out. By a series of cutting steps, the original fractures are decomposed into a large number of connected closed contours (Figure 5b) that will be used as input for the triangulation step $S_3$. The main benefit of series of cutting steps is that multiple-intersection issues vanish. For example, a problem of triple intersections between three fractures $\mathcal{F}_i, \mathcal{F}_j, \mathcal{F}_k$, is transformed to standard problems of finding intersections between $\mathcal{F}_k$ and the sub-fractures of $\mathcal{F}_i$ and $\mathcal{F}_j$. The drawback of this strategy is that it increases the number of intersection tests and, consequently, the computational time. The consequence of this feature of the method is however beyond the scope of the present paper and will be addressed in another work.

*(iii) $S_3$: Triangulation*

In this step, each sub-fracture will be meshed according to its own target edge length $h_i$ (Figure 5c). LaGriT uses a conforming Delaunay triangulation algorithm to ensure that the line of intersection between any two sub-fractures is preserved. More details can be found in Murphy et al. (2001) or Hyman et al. (2014).

*(iv) $S_4$: Merging meshed sub-fractures and post-treatment*

This step bridges the gap between the meshing part and the simulation part of our workflow. All resulting

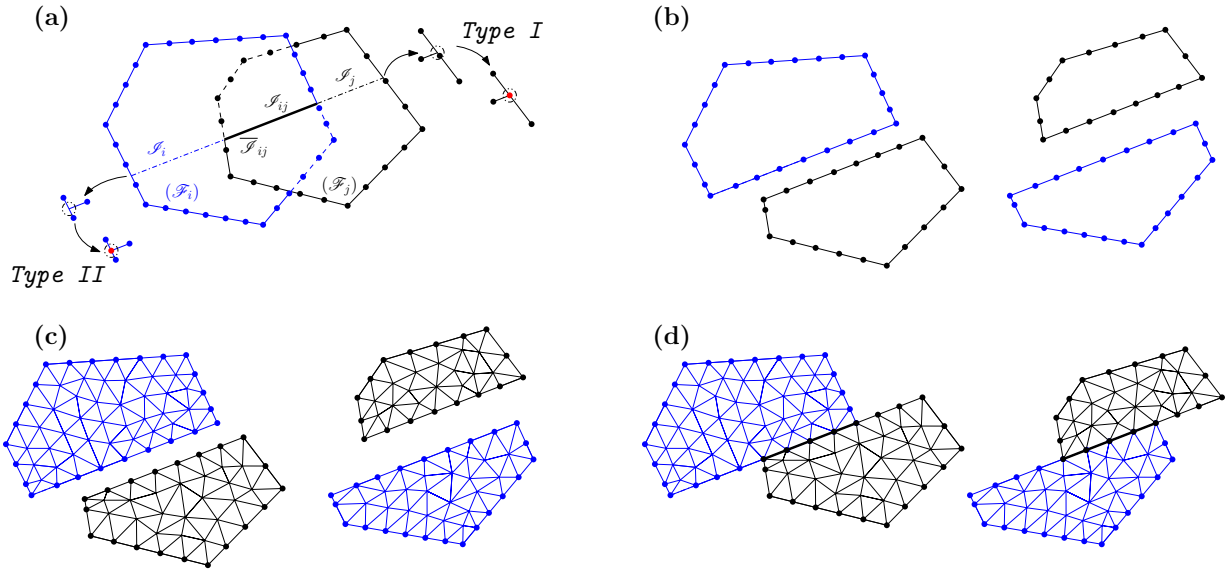



**Figure 5:** Illustration of the meshing steps: step $S_1$ – fracture intersection (a) and fracture cutting (b); step $S_3$ – polygon triangulation (c); and step $S_4$ – merging meshed polygon (d).

meshed closed contours from step $S_3$ will be merged into a single triangulation data structure, in which there will probably be duplicate vertices along the intersection lines (Figure 5d). These duplicates are detected and removed, resulting in a final fully connected mesh, that can be saved to **.inp* files which can be read by the AVS graphics program, both in binary or ASCII format. A mesh conversion of **.inp* files to **.dgf* file (DUNE Grid Format, DGF) is then required to make output mesh files compatible with the DuMux code. A detailed description of the DGF and its utilization can be found on the homepage of DUNE.

### *3.2. Meshing examples*

In this section, two applications of the FraC approach for meshing complex DFNs will be shown. The first example is a benchmark DFN, which has also been used by many previous studies (Khvoenkova and Delorme, 2011; Fourno et al., 2016), and the second one is a semi-synthetic DFN where the network's properties are closely based on the geological data from the fictional but realistic Bloemendaal reservoir (Verscheure et al., 2012).

#### *3.2.1. Benchmark 33-fracture DFN ($DFN_1$)*

The benchmark fracture network contains 33 disc-shaped fractures inside a 3 m $\times$ 3 m $\times$ 3 m cube. A common characteristic length $h$ is set for all fractures. Several meshing realizations are performed with decreasing $h$ to obtain a set of meshes with different degrees of refinement. Concretely $h$ ranges from $9.6 \times 10^{-1}$ m to $3.75 \times 10^{-3}$ m resulting in meshes that contains from $2 \times 10^3$ to $8.0 \times 10^6$ grid cells. In Figure 6, we show, as an example, the DFN mesh generated by FraC with two values of $h$: a coarse grid

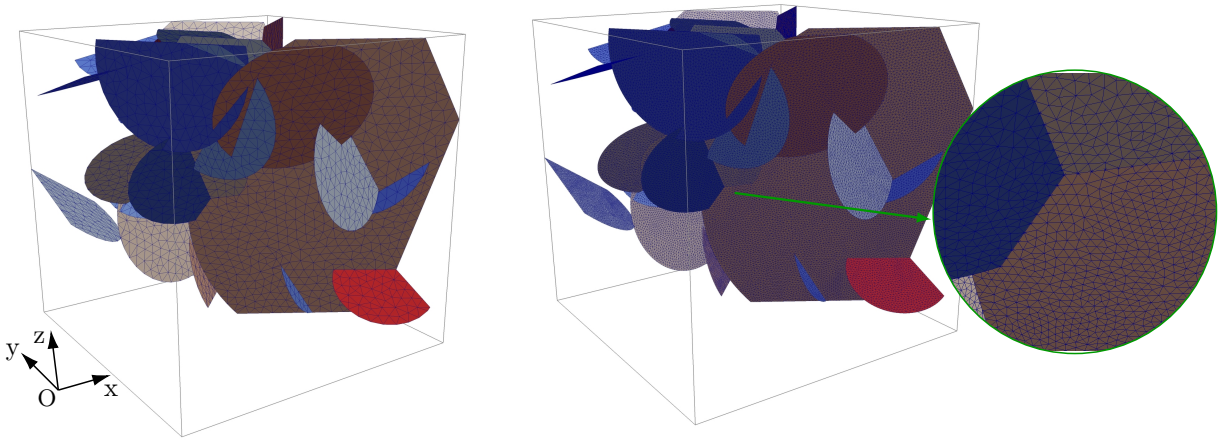



**Figure 6:** Two meshes of the benchmark DFN: a coarse mesh $\text{DFN}_{1a}$ corresponding to $h_{1a} = 0.12$ m (left) and a fine mesh $\text{DFN}_{1b}$ corresponding to $h_{1b} = 0.03$ m (right). The inset displays an example of multiple-intersection showing the mesh conformity.

$\text{DFN}_{1a}$ corresponding to $h = 0.12$ m (left) and a fine grid $\text{DFN}_{1b}$ corresponding to $h = 0.03$ m (right). This shows the ability of our code to generate not only extremely fine meshes but also very coarse meshes of the DFN while preserving the fracture network topology. The inset in this figure exhibits the mesh conformity at a multiple-intersection location.

### *3.2.2. Semi-synthetic DFN of the Bloemendaal reservoir ($DFN_2$)*

The semi-synthetic DFN consists of more than 50,000 rectangular-shaped fractures that lie on a domain of 12 km × 15 km × 1.4 km. The flowing zone depth of the reservoir is about 200 m (Figures 7a and 7b). We use the algorithm of the FracaFlow code (Beicip-FranLab, 2017) to generate the fracture network that includes two principal fracture sets: one set primarily oriented along the $x$-axis (red fractures, Figure 7a) with an average fracture spacing of 10.0 m and another primarily oriented along the $y$-axis (blue fractures, Figure 7a) with an average fracture spacing of 7.5 m. For the above colors, we refer the reader to the online version of the paper.

In practice, simulations using the DFN concept rarely model the whole fracture network of the reservoir. For example, in the oil field, the most common tests based on DFN conceptual models, such as well-tests, flow-meters or interference tests, consider the influence zone of a few kilometers around the wells (Bourbiaux et al., 2002). Therefore, in this case we focus solely on a part of the semi-synthetic DFN: only fractures inside a bounding box of $L_0 \times L_0 \times 1.4$ km (red box, Figure 7b) are taken into account, with $L_0$ initially equal to 1 km; the DFN contains around $5 \times 10^2$ fractures. An example of resulting mesh is illustrated in Figure 7c. We then expand the DFN by increasing $L_0$ to ten kilometers. The number of fractures and consequently the intersections between them inside the DFN are thus increased. Figure 7d plots a log-log graph of the variation of fracture number $N$ in the DFN bounding box with respect to the box width $L_0$, showing the relationship $N \sim L_0^2$. The unit mesh size $h$ remains constant for all meshing realizations resulting in progressively larger

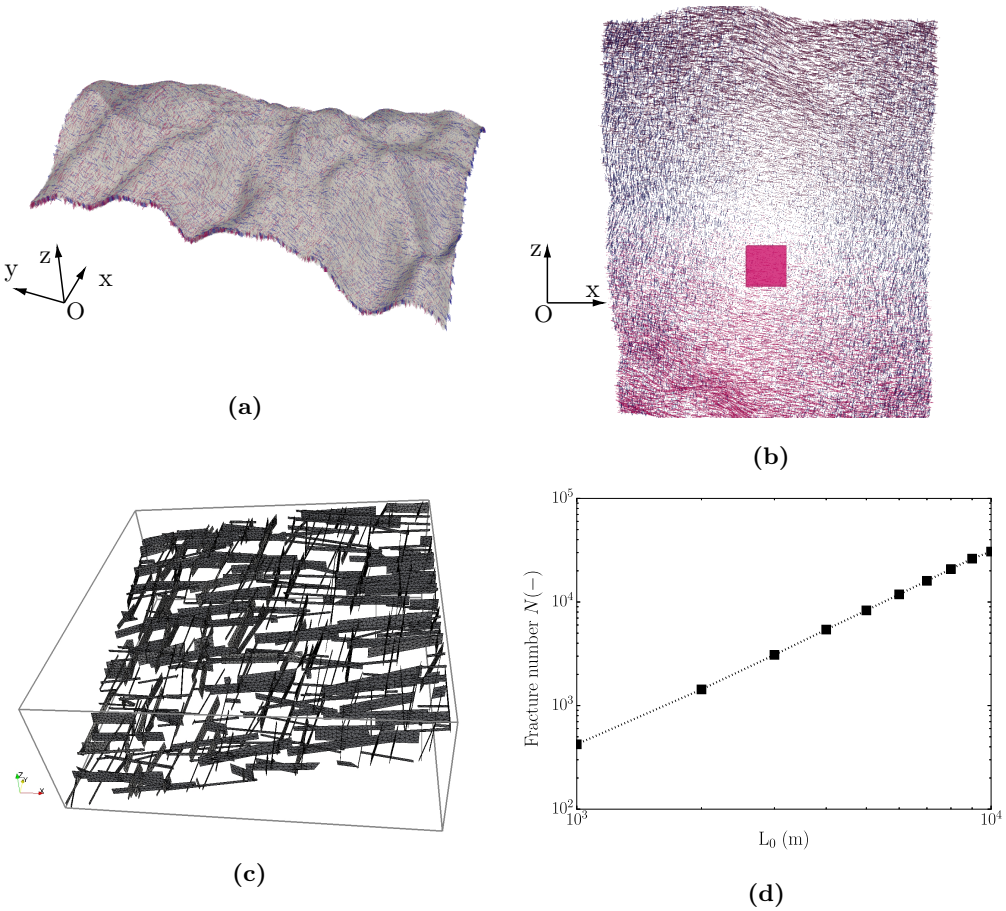



**Figure 7:** Semi-synthetic DFN: entire fracture network in lateral view (a) and in top-down view (b). The red box in (b) indicates a part of the DFN that is being meshed. An example of resulting mesh for the case $L_0 = 1$ km where the DFN contains around $5 \times 10^2$ fractures is shown in (c), and (d) reports the variation of the fracture number in the bounding box (in red) w.r.t. the box size.

grids.

## 4. Numerical results

In this section, numerical results are presented to evaluate the ability of DuMux to simulate solute transport in DFNs with increasing levels of topological complexity. Three validation tests are carried-out: (i) numerical solutions of solute transport in a single inclined fracture are compared to analytical solutions from Ogata and Banks (1961); (ii) validation is performed by comparing Dumux's simulation results to those obtained by Ahmed et al. (2015) on a simple fracture network consisting of 7 interconnected fractures; and (iii) the convergence assessment of flow and transport solutions on 33-fracture DFN meshes with increasing mesh refinement. More realistic simulations are then performed on the semi-synthetic DFN of the Bloemendaal reservoir.

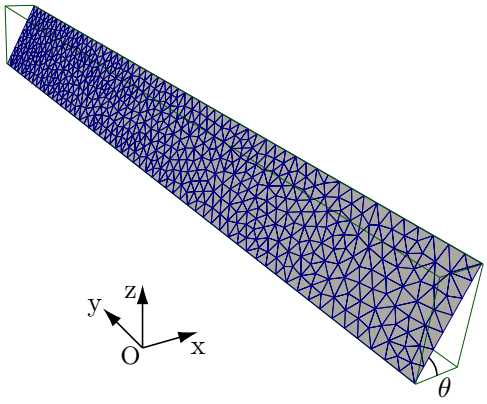



**Figure 8:** A snapshot of the mesh for a single inclined planar fracture embedded in 3D domain with $h = 0.4$m. The green rectangular cuboid indicates the bounding box around the fracture.

For simplicity, the DFNs in all examples are assumed to be homogeneous with a constant fracture aperture of $b = 10$ mm for Example 3 and $b = 1$ mm for the others. An isotropic permeability for all fractures $K = 10^3$ mD is applied for Example 2 to be consistent with that of Ahmed et al. (2015) while, for other cases, the fractures are assigned a scalar permeability tensor $K = 10^4$ mD. The fluid viscosity is constant and equals $\mu = 10^{-3}$ Pa.s.

### *4.1. Example 1: Single inclined fracture*

The single fracture is an inclined planar rectangle of length 20 m (along the $y-$axis) and of width $\sqrt{5}$ m. The angle $\theta$ between the fracture normal vector and the $z-$axis satisfies $\tan\theta = 2$.

A concentration $C(t = 0) = 0$ is applied throughout the entire domain as initial condition. Flow occurs along the $y$-axis by adopting a pressure gradient $\Delta P = 10^8$ Pa along the $y$-direction. Tracer is continuously injected into the domain with an imposed concentration $C_0$ (relative concentration $C' = {}^{C}/_{C_0} = 1$), the prescribed boundary conditions are summarized as follows:

$$\begin{cases} C = C_0 \text{ on } y = 0 \text{ m}, \\ C = 0 \text{ on } y = 20 \text{ m}, \\ \Phi = 0 \text{ otherwise.} \end{cases} \tag{5}$$

The transport analytical solution of this problem is given by Ogata and Banks (1961):

$$C(x,t) = \frac{C_0}{2}\left[erfc\left(\frac{x - vt}{2\sqrt{D_L t}}\right) + exp\left(\frac{vx}{D_L}\right) erfc\left(\frac{x + vt}{2\sqrt{D_L t}}\right)\right] \tag{6}$$

where $C_0$ is the imposed concentration on the inlet and $erfc$ is the complementary error function. When the ratio $vx/D_L$ (i.e. the Péclet number $Pe$) is large enough, the second term in equation (6) is negligibly

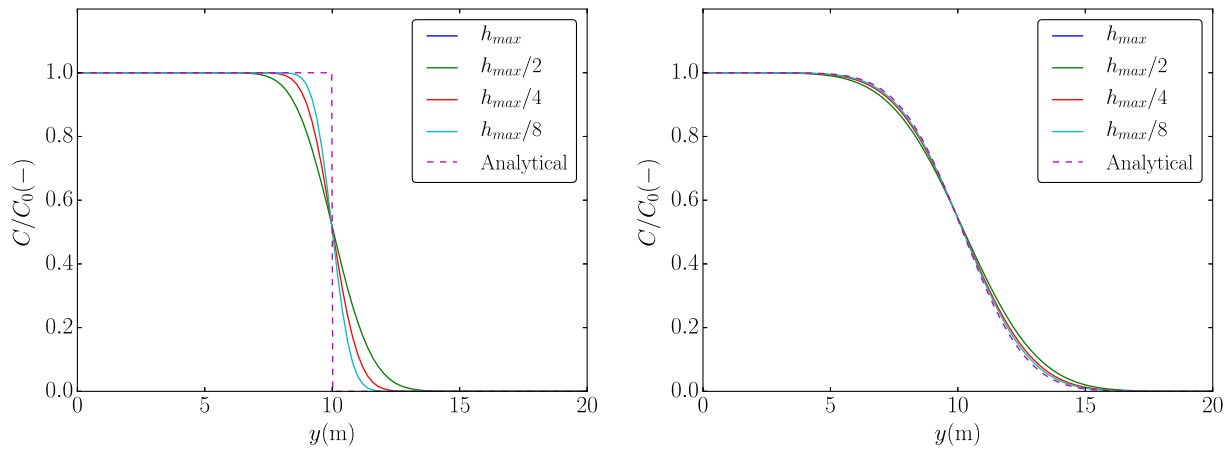



**Figure 9:** Concentration profile for the test case of continuous injection of tracer within a single fracture at $t = 2\times10^4$ (s): (a) $v_1 = 5\times10^{-4}$m.s$^{-1}$, $D_1 = 1\times10^{-9}$ m$^2$.s$^{-1}$ ($Pe_1 = 10^7$), and (b) $v_2 = 5\times10^{-4}$m.s$^{-1}$, $D_2 = 1\times10^{-4}$ m$^2$.s$^{-1}$ ($Pe_2 = 10^2$). Mesh size of the coarsest mesh $h_{max} = 0.4$m. Discontinous line on (a) represents the approximate solution from Equation (7) while the one on (b) illustrates the exact solution from Equation (6).

small and often disregarded (Ogata and Banks, 1961; Bodin et al., 2007), one obtains the approximate solution as follows:

$$C(x,t) = \frac{C_0}{2} erfc\left(\frac{x - vt}{2\sqrt{D_L t}}\right). \tag{7}$$

According to Ogata and Banks (1961), only a maximum error of less than 3% between the exact and approximate solutions in Equations (6) and (7) is shown for $Pe > 500$.

Different meshes are used to model this problem with $h$ decreasing from $h_{max} = 0.4$m, the mesh size of the coarsest mesh, to $h = h_{max}/8$. Figure 8 shows a mesh example of the fracture for $h = 0.4$m. Two cases according to two dispersion coefficient values are studied. We consider in the first case a convection-dominant process with a low dispersion coefficient $D_1 = 1\times10^{-9}$ m$^2$.s$^{-1}$ and the mean fluid velocity $v = 5\times10^{-4}$ m.s$^{-1}$ resulting in a Péclet number $Pe_1 = L\times v/D_1 = 10^7$. In the second case, $v$ remains unchanged but the dispersion coefficient is increased up to $D_2 = 1\times10^{-4}$ m$^2$.s$^{-1}$, the corresponding Péclet number is thus equal to $Pe_2 = 10^2$.

Figure 9 depicts concentration profiles obtained from numerical simulations on different meshes with variable $h$ for both cases at $t = 2\times10^4$ (s). Analytical solutions from Equation (7) for $Pe_1 = 10^7 \gg 500$ and from Equation (6) for $Pe_2 = 10^2 < 500$ are also illustrated. In the first case, there are offset between the analytical and numerical curves mainly at the front location (around $y = 10$ m). This discrepancy is possibly caused by both the error of the approximation in Equation (7) and numerical diffusion effects. It can be noted that, when the grid is refined, the numerical curves tend to the analytical solution. The numerical

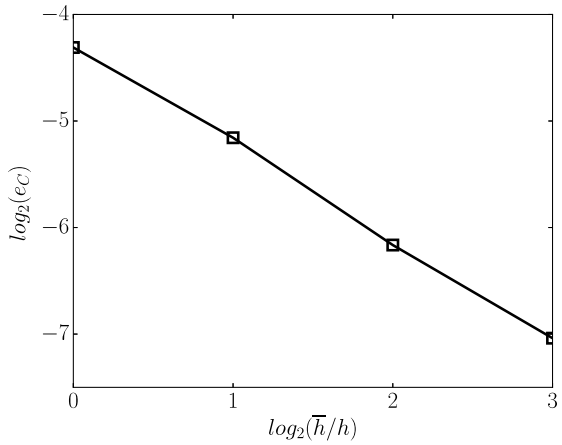



**Figure 10:** Log2-plot of the discrete concentration $L^2$ errors for the case $D_2 = 1 \times 10^{-4}$ m$^2$.s$^{-1}$ showing an order of convergence $O(h)$ for the concentration.

diffusion effect vanishes for the second case when the physical diffusion effect is stronger. Numerical solutions, particularly for the solution on the finest mesh, i.e. $h = h_{max}/8 = 0.05$ m, are in perfect agreement with the analytical one.

The exact solution from the second case $D_2 = 1 \times 10^{-4}$ m$^2$.s$^{-1}$ is now used for convergence study. Indeed, to estimate the BM convergence order, a $L^2$ norm of tracer concentration error $e_C$ is defined as:

$$e_C = \left( \frac{\Sigma_{\mathcal{B}_i \in \mathcal{T}_h} |\mathcal{B}_i| \times \left(C_{ex,\mathcal{B}_i} - C_{\mathcal{B}_i}\right)^2}{\Sigma_{\mathcal{B}_i \in \mathcal{T}_h} |\mathcal{B}_i| \times C^2_{ex,\mathcal{B}_i}} \right)^{\frac{1}{2}}. \tag{8}$$

Here, $C_{\mathcal{B}_i}$ and $C_{ex,\mathcal{B}_i}$ stand for the discrete and analytical concentration solution of the box $\mathcal{B}_i$ respectively while $|\mathcal{B}_i|$ denotes the box area.

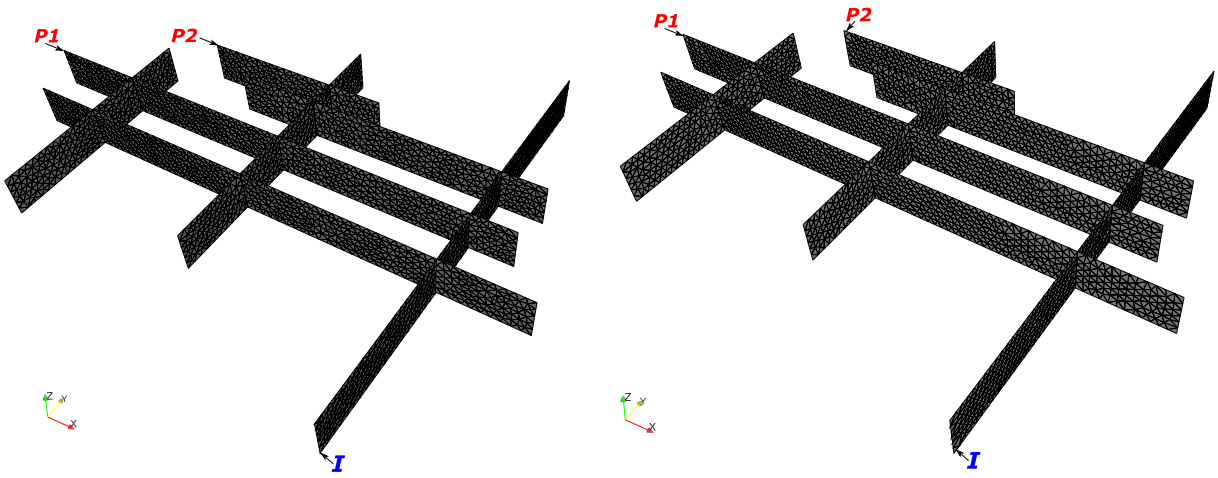
For $k \geq 2$, the convergence order $col2(k)$ of the $L^2$ norm of the error on the solution between grid level $k$ and $k-1$ can be evaluated as:

$$col2(k) = \frac{log(e_C(k)/e_C(k-1))}{log(h(k)/h(k-1))}. \tag{9}$$

Figure 10 displays the convergence rate for the concentration of the box method, showing an order of convergence of $O(h)$.

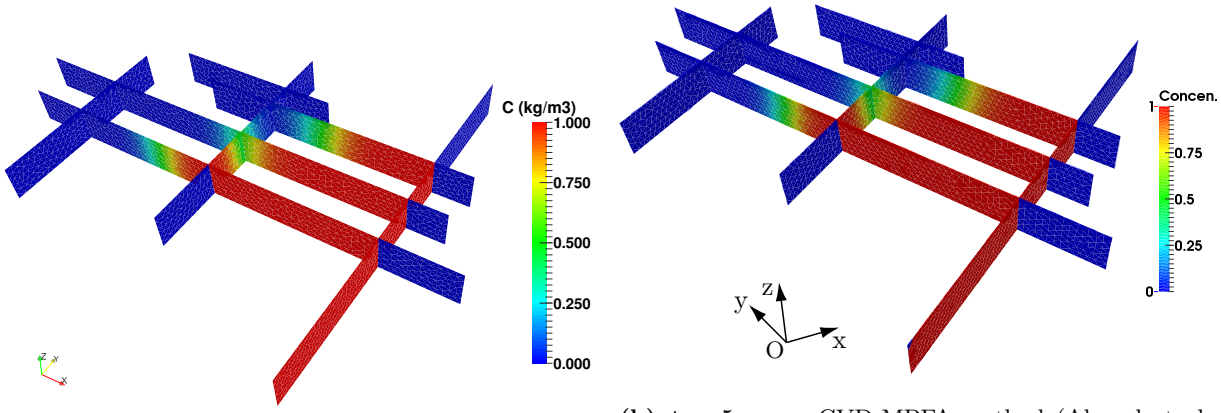
### *4.2. Example 2: 7-fracture DFN*

In this benchmark exercise, solutions of DuMux are compared to those from Ahmed et al. (2015), who use the cell-centred control-volume distributed multi-point flux approximation (CVD-MPFA) approach as discretization method. The studied DFN lies inside a 200 × 150 × 15 m$^3$ system. For validation purposes, simulation on 2D model from Ahmed et al. (2015) is chosen as a reference. Our mesh (Figure 11a) has the same resolution as that used by Ahmed et al. (2015) (figure 11b). The conforming Delaunay mesh generated by the FraC approach contains 4554 grid cells while the Ahmed's mesh contains 4662 triangular cells.



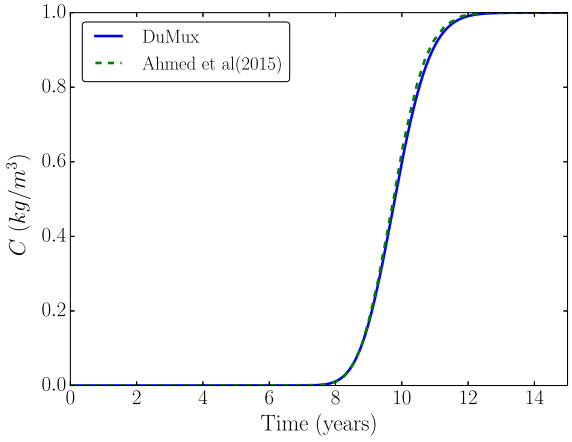

**(a)** FraC **(b)** Ahmed et al. (2015)

**Figure 11:** 7-fracture DFN: mesh generated using FraC approach (a) and that used in Ahmed et al. (2015) (b).




**(a)** $t = 5$ years, Box method (DuMux)

**(b)** $t = 5$ years, CVD-MPFA method (Ahmed et al., 2015)




**(c)** Tracer concentration w.r.t. time at producer $P_1$.

**Figure 12:** Concentration plots for the 7-fracture DFN.

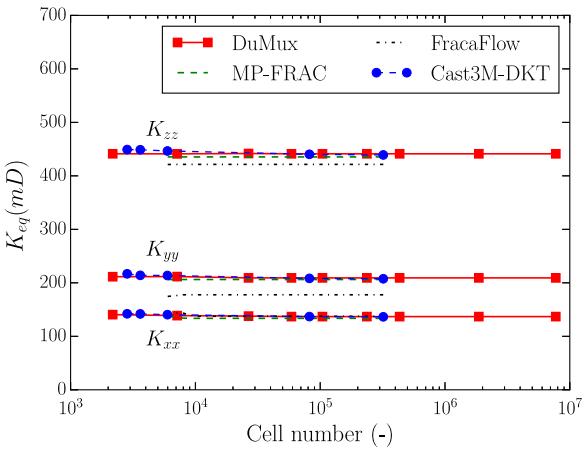



**Figure 13:** Mesh validation test of the benchmark 33-fracture DFN: equivalent permeabilities computed from the simulation results of single-phase flow at steady-state.

Figure 11 shows the locations of the injector $I$ and that of the two producers $P_1$ and $P_2$. Fluid is injected through the injector at the rate of $2.739 \times 10^{-3}$ m$^3$/day. The imposed pressure at both producers is set to 10 bar. Both gravity and diffusion effects are neglected. Figure 12a illustrates the result of DuMux for the tracer concentration $t = 5$ years, which matches perfectly with the reference result shown in Figure 12b. An excellent agreement between concentration plots at the producers $P_1$ w.r.t time in Figure 12c also validates our solutions.

### *4.3. Example 3: 33-fracture benchmark synthetic DFN*

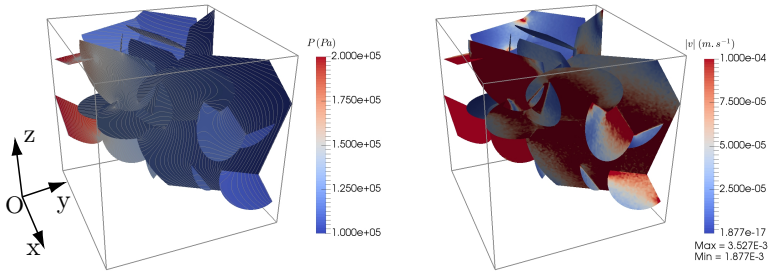
The third validation is undertaken on the $\text{DFN}_1$ based on previous studies by Fourno et al. (2013, 2016). Firstly, equivalent permeabilities are computed from the numerical solutions of steady-state single-phase flow and compared with the published results. Secondly, solute transport is modeled and analyzed from a convergence point of view.

To determine the diagonal entries of the effective permeability tensor $K_{ii}$, $i \in \{x, y, z\}$, a pressure gradient $\Delta P$ is applied on two corresponding opposite boundaries of the DFN and other boundaries are assumed to be impermeable. The equivalent permeability of a DFN can be calculated by applying the inverse Darcy's method when the fluid flow is at steady state (Zimmerman and Bodvarsson, 1996).

$$K_{ii} = \frac{\mu \times Q_{ii}}{A \times \Delta P}, \tag{10}$$

where $Q_{ii}$ is the corresponding macroscopic flux and $A$ is the cross-sectional area.

Note that the off-diagonal terms of the equivalent permeability tensor, i.e. $K_{ij}$ $(i, j \in \{x, y, z\}, i \neq j)$, can also be computed with a different setup of the boundary conditions. However, the calculation of off-diagonal equivalent permeability is out-of-scope of this work, we only investigate the principal terms $K_{xx}, K_{yy}$ and




**(a)** **(b)**

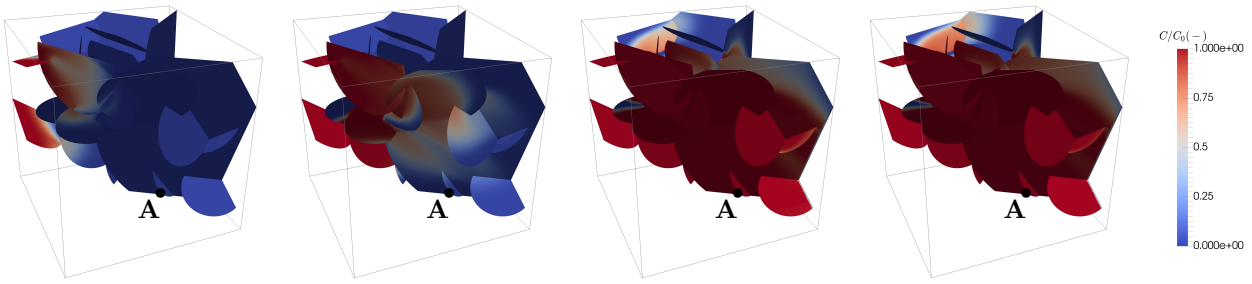
**Figure 14:** 33-fracture DFN: pressure field and iso-value lines of pressure (a) and the norm of the velocity field (b) from the simulation on a mesh of $h = h_{max} = 0.03$ m.

$K_{zz}$; the solutions will be used for validation purposes. The DFN is assumed to be homogeneous and isotropic with the intrinsic permeability is set to $10^4$ mD.

Figure 13 compares the numerical results of the equivalent permeabilities obtained from simulations with different codes including DuMux, FracaFlow (Khvoenkova and Delorme, 2011), MP-Frac (De Dreuzy et al., 2013) and Cast3M (Fourno et al., 2013). One can note that DuMux's results for the effective permeabilities along all three principal directions are similar to those obtained by MP-Frac and Cast3M. The results obtained by FracaFlow are slightly different from the others, but remain of the same order of magnitude. These discrepancies are caused by the loss of the DFN connectivity in the FracaFlow meshes. Indeed, Khvoenkova and Delorme (2011) modeled disc-shaped fractures by planar octagons, therefore some DFN intersections cannot be captured resulting in a consistent underestimation of the equivalent permeabilities along the $y$ and $z$ directions. On the contrary, it appears that the fracture connectivity in the DFN is accurately captured by FraC into resulting meshes, that are eligible for flow and transport simulations using standard simulators.

In the experiment of transport through the $\text{DFN}_1$, solute is continuously injected from an inflow boundary of the DFN. Flow occurs through the DFN along the $x$-axis by applying a pressure gradient $\Delta P = 10^5$ Pa along this axis. The dispersion coefficient is isotropic, $D = 10^{-9}$ $\text{m}^2.\text{s}^{-1}$. Simulations are run on different meshes with the grid size ranging from $1 \times 10^5$ elements (coarsest mesh) to $8 \times 10^6$ elements (finest mesh). The characteristic length $h$ varies from $h = h_{max} = 0.03$m for the coarsest mesh to $h = h_{max}/8 = 3.75 \times 10^{-3}$m for the finest mesh,

Figure 14a illustrates the pressure field and iso-value lines of the pressure while Figure 14b displays the norm of flow velocity reconstructed from numerical results using a weighting of the sub control volume velocities. In Figure 14b, blue colors represent low values (down to $1.877 \times 10^{-17}$ $\text{m.s}^{-1}$); warmer colors indicate higher values (up to $3.527 \times 10^{-3}$ $\text{m.s}^{-1}$). Color scale is set maximum at $1 \times 10^{-4}$ $\text{m.s}^{-1}$ to distinguish high- and low-flow regions. We can easily notice from Figure 14 narrowly-spaced pressure iso-value lines for high-flow regions (for instance, in fractures close to the inlet) and widely-spaced ones for low-flow areas.




**(a)** $t = 1$ hours **(b)** $t = 5$ hours **(c)** $t = 25$ hours **(d)** $t = 30$ hours

**Figure 15:** Snapshots of solute concentration distribution over time on the mesh $\text{DFN}_{1b}$ of mesh size $h = h_{max} = 0.03$ m in Figure 6. Point A illustrates an observation point located at (2.35m, 1.35m, 0m).

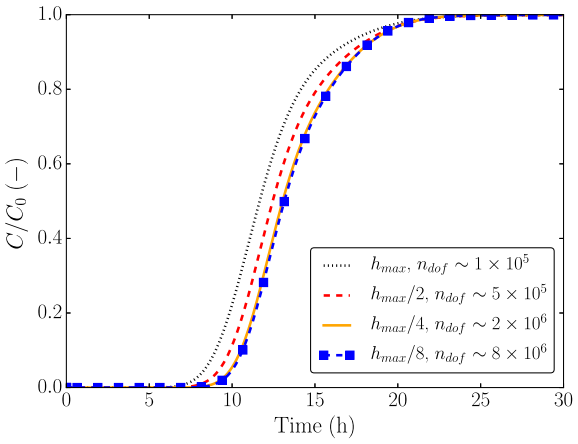



**Figure 16:** Concentration evolution at the observation point A.

Figure 15 shows the tracer concentration distribution of the numerical simulation obtained through DuMux. It reports the concentration evolution in the DFN at different timescales: after a short time of injection, $t_1 = 1$ hour (Figure 15a), an intermediate time $t_2 = 5$ hours (Figure 15b) and a relatively long time $t_3 = 25$ hours (Figure 15c) and $t_4 = 30$ hours (Figure 15d) when the solute has reached the outflow boundary. We notice that for long times, i.e. after 25 hours, the solute appears nearly in the whole DFN; in addition, the solute displacement occurs slowly in the DFN, which is expressed through the similar solute distributions on Figures 15c and 15d. Clearly, this slow motion is the result of both low diffusion effect and low velocity: the distribution of low concentration areas fits correctly with low-flow regions in Figure 14b.

Figure 16 exhibits the concentration evolution at the observation point A of coordinate (2.35m, 1.35m, 0m) obtained from simulations run on 4 grids, the mesh refinement ratio is a factor of 2. Obviously, the arrival time for coarse meshes is earlier than for fine meshes due to numerical diffusion. The observed curves obtained from the finest meshes, i.e. the third and fourth grid levels, overlap each other (discrepancy less than 1%). For instance, the relative concentration at this point at $t = 15$h for the two coarsest meshes is about 0.85 and 0.79 respectively, and for both two finest mesh is roughly equal to 0.73. Through this figure, the convergence of the solution with respect to mesh refinement is also judged.

### *4.4. Example 4: realistic DFN of the Bloemendaal reservoir*

We study in the last example solute transport within a complex realistic DFN of the Bloemendaal reservoir with $L_0 = 3$ km. The network contains 3088 fractures and the mesh consists of about 1.1 million grid cells. As mentioned above, the fracture network is well connected and seems to be homogenizable.

Single-phase flow simulations to compute effective permeabilities of the DFN block as described in subsection 4.3 are also carried out, giving $K_{xx} = 4.58 \times 10^{-18}$ m$^2$ and $K_{yy} = 3.0 \times 10^{-18}$ m$^2$. It is worth noticing that, although there are more $y$-axis-oriented fractures than $x$-axis-oriented ones within the DFN, the effective permeability of DFN along the $x$-axis is surprisingly about 1.5 times greater than the $y$-axis one. This can be explained by the fact that the fracture network is more connected and therefore more conductive in the $x$-direction than in the $y$-direction.

Fluid is injected into the domain through an injection well located in the middle of the DFN, the injection rate is $5 \times 10^{-2}$ m$^3$/day. A Dirichlet condition $P = 10$ bar is applied for all lateral boundaries while a no-flow Neumann condition is applied for top and bottom boundaries of the domain. As in Example 3, we consider the case where the dispersion coefficient is isotropic, $D = 10^{-9}$ m$^2$.s$^{-1}$. Figure 17 shows solute concentration distribution at $t = 10$ years (Figure 17a), $t = 100$ years (Figure 17b) and $t = 250$ years (Figure 17c). We note from these figures that there are low-concentration regions which exist locally within the solute plume. However, disregarding local phenomena, the DFN behaves as a continuous homogeneous medium at macroscopic scale: the concentration distribution has an elliptical form.

In practice, the overall development of a solute plume within a 2D or 3D heterogeneous medium may be precisely measured through a moment analysis. This approach is an appropriate tool for examining spatial

effects of the domain on developing solute plumes, for identifying loss of mass during flow process and for assessing large-scale dispersive behavior of the flowing domain itself (Freyberg, 1986; Tompson and Gelhar, 1990).

Let us remind that the $ijk$-th moment of the concentration distribution in space, $M_{ijk}$ is defined as (Aris, 1956)

$$M_{ijk}(t) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \omega C(x,y,z,t) x^i y^j z^k dxdydz. \tag{11}$$

where $\omega$ [-] and $C$ [-] denote the porosity and the relative tracer concentration, respectively. The zeroth-order moment $M_{000}(t)$ is the total mass of the solute in the domain. The first-order moment about the origin, normalized by $M_{000}$, describes the propagation of the center of mass $\mathrm{O}_c(x_c(t), y_c(t), z_c(t))$ of the solute plume:

$$x_c(t) = M_{100}/M_{000}\ ;\ y_c(t) = M_{010}/M_{000}\ ;\ z_c(t) = M_{001}/M_{000}. \tag{12}$$

The centered second-order tensor defines a spatial covariance tensor reflecting the spreading of mass about $\mathrm{O}_c$:

$$\overline{\overline{\sigma}} = \begin{bmatrix} \sigma_{xx} & \sigma_{xy} & \sigma_{xz} \\ \sigma_{yx} & \sigma_{yy} & \sigma_{yz} \\ \sigma_{zx} & \sigma_{zy} & \sigma_{zz} \end{bmatrix} \tag{13}$$

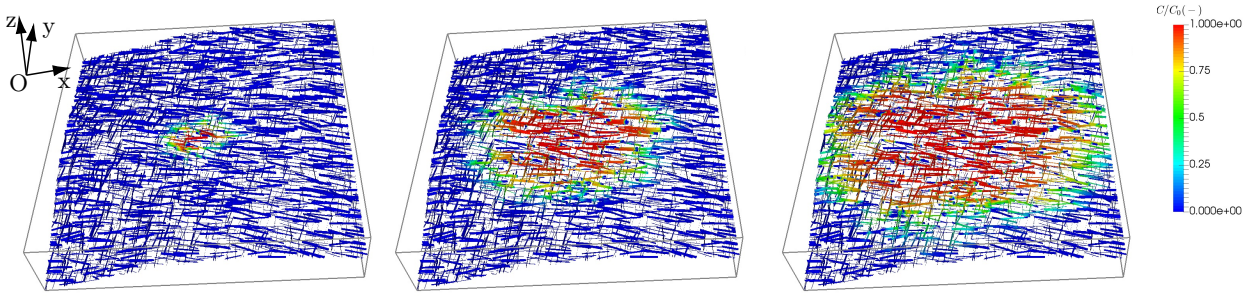
where

$$\begin{aligned} \sigma_{xx} = \frac{M_{200}}{M_{000}} - x_c^2,\ \sigma_{yy} = \frac{M_{200}}{M_{000}} - y_c^2,\ \sigma_{zz} = \frac{M_{200}}{M_{000}} - z_c^2, \\ \sigma_{xy} = \sigma_{yx} = \frac{M_{110}}{M_{000}} - x_c y_c, \\ \sigma_{xz} = \sigma_{zx} = \frac{M_{101}}{M_{000}} - x_c z_c, \\ \sigma_{yz} = \sigma_{zy} = \frac{M_{011}}{M_{000}} - y_c z_c. \end{aligned}$$

Diagonalization of $\overline{\overline{\sigma}}$ provides the orientation of principal axes $(x', y', z')$ as well as the eigenvalues of the spatial covariance tensor $\sigma_{x'x'}, \sigma_{y'y'}, \sigma_{z'z'}$.

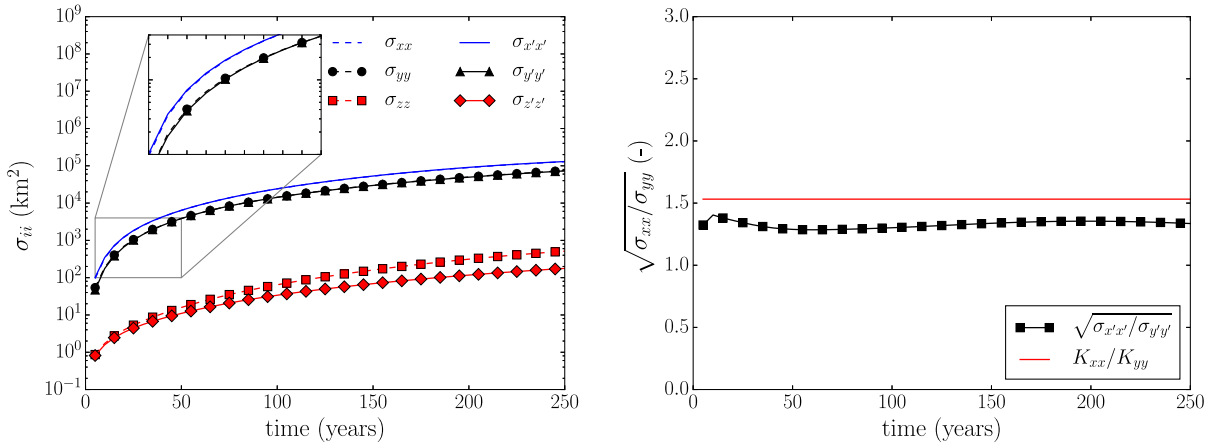
In addition, the derivative over time of the covariance is proportional to the dispersion tensor (Aris, 1956; Freyberg, 1986), one could expect that the square root of the ratio between the eigenvalues of the centered second-order moments, $\sqrt{\sigma_{x'x'}/\sigma_{y'y'}}$, reflects the anisotropy of the elliptic solute plume for the homogenized media.

Figure 18a reports the variation versus time of the centered second-order moments $\sigma_{xx}, \sigma_{yy}, \sigma_{zz}$ and that of the eigenvalues of the $\overline{\overline{\sigma}}$ tensor, i.e. $\sigma_{x'x'}, \sigma_{y'y'}, \sigma_{z'z'}$. We note that along the $x$- and $y$-axis, the centered second-order moments and the corresponding eigenvalue match correctly, only a difference between $\sigma_{zz}$ and $\sigma_{z'z'}$ is observed. However, since both $\sigma_{zz}$ and $\sigma_{z'z'}$ are roughly two orders of magnitude less than those along the $x$- and $y$- axis, changing of the $z'$-axis with respect to the $z$-axis can also be negligible. This proves




**(a)** $t = 10$ years **(b)** $t = 100$ years **(c)** $t = 250$ years

**Figure 17:** Solute concentration distribution over time on a mesh of $L_0 = 3$ km.




**(a)** **(b)**

**Figure 18:** Temporal variation of $\sigma_{ii}$ $(i \in x, y, z)$ (a) and ratio between effective absolute permeabilities $K_{xx}/K_{yy}$ as well as the ratio between the centered second-order moment $\sqrt{\sigma_{x'x'}/\sigma_{y'y'}}$ (b).

that the principal axes for the developing direction of the solute plume $x', y', z'$ and the original orientations $x, y, z$ nearly coincide.

Figure 18b illustrates the time-evolution of the $\sqrt{\sigma_{x'x'}/\sigma_{y'y'}}$ ratio. One can notice that $\sqrt{\sigma_{x'x'}/\sigma_{y'y'}}$ is fairly correlated with that of the effective permeabilities $K_{xx}/K_{yy}$; a discrepancy exists between them but it is still acceptable and generally remains stable over time. At early times when the amount of solute in the domain remains low, $\sqrt{\sigma_{x'x'}/\sigma_{y'y'}}$ is a little greater than the average value because local fracture connectivity could have significant effects on the ensemble solute distribution. After 50 years, local effects vanish and the curve becomes relatively stable until 100 years. However, $\sqrt{\sigma_{x'x'}/\sigma_{y'y'}}$ increases slightly again after 150 years because the solute has already reached lateral boundaries of the DFN.

## 5. Summary and discussion

The purpose of this work is to obtain high-accuracy simulations of solute transport within large, three-dimensional Discrete Fracture Networks (DFNs). A workflow is described, including both the mesh generation for DFNs using the conforming mesh method FraC and the simulation framework using the DuMux code. The meshing approach FraC, initially developed by Fourno et al. (2016), relies on the idea of decomposing fractures along extended intersection segments between fractures to obtain a set of interconnected closedW contours, that will then be discretized with the same point density on shared segments between them to ensure the conformity of the final mesh. Conforming meshes were then used for transport simulations with DuMux using a vertex-centered discretization method, or box method. A number of tests were conducted for both validation and demonstration purposes. Validation tests gave satisfactory results: (i) numerical solutions of solute transport on a DFN of single inclined fracture are in agreement with analytical solutions by Ogata and Banks (1961), an a-posteriori convergence analysis showed a convergence rate of $O(h)$ for concentration, where $h$ is the mesh size; (ii) results on 7-fracture DFN obtained by DuMux using the box method match perfectly with those obtained by the cell-centred control-volume distributed multi-point flux approximation (CVD-MPFA) method (Ahmed et al., 2015); and (iii) simulations on a benchmark 33-fracture DFN showed convergence of solution with increasing mesh refinement. Finally, we considered development of solute plume injected into a DFN produced from the Bloemendaal reservoir data that consists of thousands of fractures. This test case reveals that our method is able to deal with large-scale DFN containing a moderate fracture number.

High-accuracy simulations obtained through this work appear to be very useful for validation of earlier theoretical studies on simplified models, for instance pipe-networks models (Noetinger and Jarrige, 2012; Noetinger, 2015), as well as for benchmarking purposes to test the reliability and performance of commercial or academic simulators. These results also motivate many directions for further works, both for the mesh generation and at the numerical stage. Objectively speaking, numerical simulations on large grids using DuMux could lead to prohibitive CPU times. In order to overcome this issue, several measures could be examined, either adopting the adaptive mesh refinement (AMR) approach (Sander et al., 2015) or distributing the workload on different resources by using parallel computing.

It has also been shown that one can only run massively parallel simulations in DuMux if the DUNE grid manager allows it. Nevertheless, to the best of the authors' knowledge, DUNE::FoamGrid is not yet parallelizable. On the contrary, the AMR approach is supported by DUNE::FoamGrid, i.e. simulations can be run on time-evolving meshes with local mesh refinement/coarsening inside DuMux. On the other hand, the AMR approach can be included into FraC frameworks itself without major difficulties. In principle, only vicinity areas of intersections should be refined and a coarse meshing is applied for the remaining. This work can easily be carried out within LaGriT as in the dfnWorks framework (Hyman et al., 2015).

Furthermore, although all DFNs in this paper are assumed to be homogeneous for simplicity, the framework is, however, able to handle heterogeneity. More precisely, the step $S_4$ of the FraC approach provides

flexibility for both fracture-to-fracture and in-fracture heterogeneity. Heterogeneous parameters such as fracture aperture, transmissivity, permeability and porosity are assigned to each mesh element. In the case of fracture-to-fracture heterogeneity, grid cells in a same fracture are assigned a common material index `matID`, the heterogeneous parameters are then distributed according to `matID`. To model inside-fracture heterogeneity, spatially correlated random fields for fracture aperture are first generated using the R package gstat (Pebesma, 2004). The correlation length of random fields $\lambda$ is proportional to fracture size. Sequential unconditional Gaussian simulations are carried out to generate random fields for fracture aperture constrained by a simple kriging variogram. The permeability and transmissivity fields are then derived from the aperture distribution using the well-known transmissivity-aperture cubic law (Adler et al., 2012). Heterogeneous data are then directly attached into **.dgf* grid files and will be read in the DuMux via the DUNE grid interface `DUNE-grid` (Bastian et al., 2008). Preliminary results show that simulations on heterogeneous DFN are in general more time-consuming than on homogeneous ones, hence once again, this highlights the importance of parallel computing.

In addition, for the purpose of various applications, more sophisticated problems including, for instance, transient single-phase flow, multi-phase multi-component flow, sorption or reactive transport could also be examined.

Finally, another direction to extend our works is to account for the matrix to fracture flow. That may pose challenges for both the mesh generation and simulation procedure. For instance, meshing dense-distributed DFNs with matrix could yield poor quality tetrahedrons that would dramatically degrade numerical performances. Furthermore, it requires advanced discretization methods able to deal with the exchange at matrix-fracture interfaces. For that, Vertex Approximate Gradient (VAG) Scheme (Brenner et al., 2015) or cell-centred control-volume distributed multi-point flux approximation (CVD-MPFA) method (Ahmed et al., 2015) could be excellent candidates.

**Acknowledgements**

The authors would like to thank the Computational Earth Science Group of LANL for providing us the source code of LaGriT and dfnWorks.

*Counts of words*

6560+78+569 (19/18/374/13) Total text+headers+captions

7594 words

*Counts of characters*

33378+520+2839 (19/18/374/13) Total text+headers+captions

37124 Characters

## References


Adler, P.M., Thovert, J.F., Mourzenko, V.V., 2012. Fractured porous media. Oxford University Press.

Ahmed, R., Edwards, M.G., Lamine, S., Huisman, B.A., Pal, M., 2015. Three-dimensional control-volume distributed multi-point flux approximation coupled with a lower-dimensional surface fracture model. Journal of Computational Physics 303, 470–497.

Aquilina, L., Rose, P., Vaute, L., Brach, M., Gentier, S., Jeannot, R., Jacquot, E., Audigane, P., Tran-Viet, T., Jung, R., et al., 1998. A tracer test at the Soultz-Sous-Forets Hot Dry Rock geothermal site, in: Proceedings, Twenty-Third Workshop on Geothermal Reservoir Engineering, Stanford University, Stanford, California, pp. 343–350.

Aris, R., 1956. On the dispersion of a solute in a fluid flowing through a tube, in: Proceedings of the Royal Society of London A: mathematical, physical and engineering sciences, The Royal Society. pp. 67–77.

Bastian, P., Blatt, M., Dedner, A., Engwer, C., Klöfkorn, R., Kornhuber, R., Ohlberger, M., Sander, O., 2008. A generic grid interface for parallel and adaptive scientific computing. Part II: Implementation and tests in DUNE. Computing 82, 121–138.

Bear, J., Tsang, C.F., De Marsily, G., 2012. Flow and contaminant transport in fractured rock. Academic Press.

Beicip-FranLab, 2017. FracaFlow. URL: `http://www.beicip.com/fractured-reservoirs`.

Benedetto, M.F., Berrone, S., Pieraccini, S., Scialò, S., 2014. The virtual element method for discrete fracture network simulations. Computer Methods in Applied Mechanics and Engineering 280, 135–156.

Benedetto, M.F., Berrone, S., Scialò, S., 2016. A globally conforming method for solving flow in discrete fracture networks using the virtual element method. Finite Elements in Analysis and Design 109, 23–36.

Berrone, S., Pieraccini, S., Scialo, S., 2013. A pde-constrained optimization formulation for discrete fracture network flows. SIAM Journal on Scientific Computing 35, B487–B510.

Berrone, S., Pieraccini, S., Scialò, S., 2017. Non-stationary transport phenomena in networks of fractures: effective simulations and stochastic analysis. Computer Methods in Applied Mechanics and Engineering 315, 1098–1112.

Bodin, J., Porel, G., Delay, F., Ubertosi, F., Bernard, S., De Dreuzy, J.R., 2007. Simulation and analysis of solute transport in 2d fracture/pipe networks: The solfrac program. Journal of contaminant hydrology 89, 1–28.

Bourbiaux, B., 2010. Fractured reservoir simulation: A challenging and rewarding issue. Oil & Gas Science and Technology–Revue de l'Institut Français du Pétrole 65, 227–238.

Bourbiaux, B., Basquet, R., Cacas, M.C., Daniel, J.M., Sarda, S., et al., 2002. An integrated workflow to account for multi-scale fractures in reservoir simulation models: implementation and benefits, in: Abu Dhabi International Petroleum Exhibition and Conference, Society of Petroleum Engineers.

Brenner, K., Groza, M., Guichard, C., Masson, R., 2015. Vertex approximate gradient scheme for hybrid dimensional two-phase darcy flows in fractured porous media. ESAIM: Mathematical Modelling and Numerical Analysis 49, 303–330.

Cacas, M.C., Ledoux, E., de Marsily, G., Tillie, B., Barbreau, A., Durand, E., Feuga, B., Peaudecerf, P., 1990. Modeling fracture flow with a stochastic discrete fracture network: calibration and validation: 1. The flow model. Water Resources Research 26, 479–489. URL: http://dx.doi.org/10.1029/WR026i003p00479, doi:10.1029/WR026i003p00479.

Cast3M, 2017. Website for the computer code cast3m. URL: http://www.cast3m.cea.fr.

De Dreuzy, J.R., Pichot, G., Poirriez, B., Erhel, J., 2013. Synthetic benchmark for modeling flow in 3D fractured media. Computers & Geosciences 50, 59–71.

Delorme, M., Bossie-Codreanu, D., Ben-Gharbia, I., Khebzegga, O., Khebzegga, N., Ricois, O., et al., 2016. Unconventional production forecast needs integration of field hydraulic stimulation data through fracture model calibration and optimized numerical scheme, in: SPE Argentina Exploration and Production of Unconventional Resources Symposium, Society of Petroleum Engineers.

Dershowitz, W., Fidelibus, C., 1999. Derivation of equivalent pipe network analogues for three-dimensional discrete fracture networks by the boundary element method. Water Resources Research 35, 2685–2691.

DuMux, 2017. Web site for the computer code DuMux. URL: http://www.dumux.org/.

Erhel, J., De Dreuzy, J.R., Poirriez, B., 2009. Flow simulation in three-dimensional discrete fracture networks. SIAM Journal on Scientific Computing 31, 2688–2705.

Flemisch, B., Darcis, M., Erbertseder, K., Faigle, B., Lauser, A., Mosthaf, K., Müthing, S., Nuske, P., Tatomir, A., Wolff, M., et al., 2011. Dumu$^x$: DUNE for multi-{phase, component, scale, physics,...} flow and transport in porous media. Advances in Water Resources 34, 1102–1112.

Fourno, A., Benabderrahmane, H., Grenier, C., Delay, F., 2007. Development and qualification of a smeared fracture modelling approach for transfers in fractured media, in: Groundwater in Fractured Rocks: IAH Selected Paper Series, volume 9. Taylor & Francis, pp. 587–603.

Fourno, A., Grenier, C., Benabderrahmane, A., Delay, F., 2013. A continuum voxel approach to model flow in 3d fault networks: A new way to obtain up-scaled hydraulic conductivity tensors of grid cells. Journal of Hydrology 493, 68–80.

Fourno, A., Noetinger, B., Borderie, C., 2016. An original and useful approach to mesh a discrete fracture network using a delaunay triangulation: Application on flow and transport upscaling from characterization scale to reservoir scale, in: Computational Methods in Water Resources Conference 2016, XXI International Conference, pp. 36–36.

Freyberg, D.L., 1986. A natural gradient experiment on solute transport in a sand aquifer: 2. spatial moments and the advection and dispersion of nonreactive tracers. Water Resources Research 22, 2031–2046.

Grenier, C., Fourno, A., Mouche, E., Delay, F., Benabderrahmane, H., 2005. Assessment of Retention Processes for Transport in a Fractured System at Äspö (Sweden) Granitic Site: from Short-Time Experiments to Long-Time Predictive Models. Dynamics of Fluids and Transport in Fractured Rock , 117–128.

Gylling, B., Moreno, L., Neretnieks, I., 1999. The channel network model—a tool for transport simulations in fractured media. Groundwater 37, 367–375.

Hammond, G.E., Lichtner, P.C., Mills, R.T., 2014. Evaluating the performance of parallel subsurface simulators: An illustrative example with PFLOTRAN. Water Resources Research 50, 208–228.

Huang, N., Jiang, Y., Li, B., Liu, R., 2016. A numerical method for simulating fluid flow through 3-d fracture networks. Journal of Natural Gas Science and Engineering 33, 1271–1281.

Huber, R., Helmig, R., 2000. Node-centered finite volume discretizations for the numerical simulation of multiphase flow in heterogeneous porous media. Computational Geosciences 4, 141–164.

Hyman, J.D., Gable, C.W., Painter, S.L., Makedonska, N., 2014. Conforming delaunay triangulation of stochastically generated three dimensional discrete fracture networks: a feature rejection algorithm for meshing strategy. SIAM Journal on Scientific Computing 36, A1871–A1894.

Hyman, J.D., Karra, S., Makedonska, N., Gable, C.W., Painter, S.L., Viswanathan, H.S., 2015. dfnWorks: A discrete fracture network framework for modeling subsurface flow and transport. Computers & Geosciences 84, 10–19.

Jourdain, X., Colliat, J.B., De Sa, C., Benboudjema, F., Gatuingt, F., 2014. Upscaling permeability for fractured concrete: meso–macro numerical approach coupled to strong discontinuities. International Journal for Numerical and Analytical Methods in Geomechanics 38, 536–550.

Karimi-Fard, M., Gong, B., Durlofsky, L.J., 2006. Generation of coarse-scale continuum flow models from detailed fracture characterizations. Water resources research 42.

Karra, S., Makedonska, N., Viswanathan, H.S., Painter, S.L., Hyman, J.D., 2015. Effect of advective flow in fractures and matrix diffusion on natural gas production. Water Resources Research 51, 8646–8657.

Khvoenkova, N., Delorme, M., 2011. An optimal method to model transient flows in 3D discrete fracture network, in: IAMG conference, pp. 1238–1249.

Lichtner, P.C., Hammond, G.E., Lu, C., Karra, S., Bisht, G., Andre, B., Mills, R.T., Kumar, J., 2013. PFLOTRAN User Manual. Technical Report.

Long, J., Remer, J., Wilson, C., Witherspoon, P., 1982. Porous media equivalents for networks of discontinuous fractures. Water Resources Research 18, 645–658.

Los Alamos Grid Toolbox, 2013. Los Alamos National Laboratory. URL: `http://lagrit.lanl.gov`.

Makedonska, N., Painter, S.L.L., Bui, Q.M., Gable, C.W., Karra, S., 2015. Particle tracking approach for transport in three-dimensional discrete fracture networks. Computational Geosciences 19, 1123–1137.

Matthai, S.K., Nick, H.M., 2009. Upscaling two-phase flow in naturally fractured reservoirs. AAPG bulletin 93, 1621–1632.

Murphy, M., Mount, D.M., Gable, C.W., 2001. A point-placement strategy for conforming delaunay tetrahedralization. International Journal of Computational Geometry & Applications 11, 669–682.

Neuman, S., 1988. A proposed conceptual framework and methodology for investigating flow and transport in swedish crystalline rocks. SKB Swedish Nuclear Fuel and Waste Management Co., Stockholm, September, Arbetsrapport , 88–37.

Noetinger, B., 2015. A quasi steady state method for solving transient Darcy flow in complex 3D fractured networks accounting for matrix to fracture flow. Journal of Computational Physics 283, 205–223.

Noetinger, B., Jarrige, N., 2012. A quasi steady state method for solving transient Darcy flow in complex 3D fractured networks. Journal of Computational Physics 231, 23–38.

Ogata, A., Banks, R.B., 1961. A solution of the differential equation of longitudinal dispersion in porous media: fluid movement in earth materials. US Government Printing Office.

Pebesma, E.J., 2004. Multivariable geostatistics in S: the gstat package. Computers & Geosciences 30, 683–691.

Pichot, G., Erhel, J., De Dreuzy, J.R., 2010. A mixed hybrid mortar method for solving flow in discrete fracture networks. Applicable Analysis 89, 1629–1643.

Pichot, G., Erhel, J., de Dreuzy, J.R., 2012. A generalized mixed hybrid mortar method for solving flow in stochastic discrete fracture networks. SIAM Journal on scientific computing 34, B86–B105.

Sander, O., Koch, T., Schröder, N., Flemisch, B., 2015. The Dune FoamGrid implementation for surface and network grids. CoRR abs/1511.03415. URL: `http://arxiv.org/abs/1511.03415`.

Svensson, U., 2001. A continuum representation of fracture networks. Part I: Method and basic test cases. Journal of Hydrology 250, 170–186.

588 Tompson, A.F., Gelhar, L.W., 1990. Numerical simulation of solute transport in three-dimensional, randomly
589 heterogeneous porous media. Water Resources Research 26, 2541–2562.

590 Verscheure, M., Fourno, A., Chilès, J.P., 2012. Joint inversion of fracture model properties for co2 storage
591 monitoring or oil recovery history matching. Oil & Gas Science and Technology–Revue d'IFP Energies
592 nouvelles 67, 221–235.

593 Zimmerman, R., Bodvarsson, G., 1996. Effective transmissivity of two-dimensional fracture networks, in:
594 International Journal of Rock Mechanics and Mining Sciences & Geomechanics Abstracts, Elsevier. pp.
595 433–438.