MOX-Report No. 33/2015

An upscaling procedure for fractured reservoirs with non-matching grids

Fumagalli, A; Pasquale, L; Zonca, S.; Micheletti, S.

MOX, Dipartimento di Matematica
Politecnico di Milano, Via Bonardi 9 - 20133 Milano (Italy)

mox-dmat@polimi.it http://mox.polimi.it

An upscaling procedure for fractured reservoirs with non-matching grids

Alessio Fumagalli[♯] Luca Pasquale[†] Stefano Zonca[♯]
Stefano Micheletti[♯]

September 28, 2015

[†] MOXOFF S.r.l., Milan, Italy luca.pasquale@moxoff.com

[♯] MOX– Modellistica e Calcolo Scientifico
Dipartimento di Matematica "F. Brioschi"
Politecnico di Milano
via Bonardi 9, 20133 Milano, Italy
alessio.fumagalli@polimi.it
stefano.zonca@polimi.it
stefano.micheletti@polimi.it

Abstract

Upscaling of geological models for reservoir simulation is an active and important area of research. In particular, we are interested in reservoirs where the rock matrix exhibits an intricate network of fractures, which usually act as a preferential path to the flow. Accounting for fractures' contribution in the simulation of a reservoir is of paramount importance. Here, we have focused on obtaining effective parameters (*e.g.* transmissibility) on a $3D$ computational grid on the reservoir scale, that account for the presence, at a finer spatial scale, of fractures, and network of fractures. We have, essentially, followed the idea illustrated in [24], yet this work has some notable aspects of innovation in the way the procedure has been implemented, and in its capability to consider rather general corner-point grids, like the ones normally used in reservoir simulations in the industry, and complex and realistic fracture networks. In particular, novel contribution is the employment of EDFM for computing fracture-fracture and matrix-fracture transmissibilities, with a remarkable gain in speed-up. The output is in form of transmissibility that can be used for reservoir simulations with software like Eclipse, Intersect, or GPRS. The results demonstrate the effectiveness and computational efficiency of the numerical procedure, and of the developed software, which is now ready for further testing and industrialization.

1 Introduction

Reservoir simulation is widely applied in the petroleum industry for the prediction and management of reservoir performance. A primary input to the flow simulator is the geological description of the reservoir, which is typically in the form of a high resolution geocellular model containing petrophysical (*e.g.* porosity and permeability) data [11]. Geocellular models are commonly built as geostatistical realizations constrained to data of different types and scales. These models are often generated at high levels of resolution because it is well established that fine scale features can impact reservoir performance significantly. Since the high level of detail of the geocellular models significantly exceeds the capabilities of standard reservoir simulators, procedures are required to coarsen the reservoir description to scales more suitable for flow computation. Such procedures are commonly referred to as "upscaling" or "scale up" techniques. There are a variety of different approaches to upscaling geological models for reservoir simulation. A key issue with any upscaling procedure is how well the coarsened (upscaled) model replicates important aspects of the fine scale flow behaviour, such as total injection or production rate, average pressure or saturation throughout the reservoir, and breakthrough times of injected fluids.

The accurate modeling of flow through complex networks of thousands of intersecting fractures in a reservoir is important for many types of problems related to the oil industry. Considering the scientific literature, there are, however, a large number of challenges associated with predicting the flow through fractured systems. An example is the development of reliable and efficient mathematical models to describe the fractures networks from their geological representation and the flow through those networks and the rock matrix. Since the permeability of the fractures is typically orders of magnitudes greater than that of the rock matrix, water, oil, and gas flow preferably in networks of fractures. An efficient and effective modeling of fluid flows in fractured media is a complex task due to this highly heterogeneous nature of the flow processes. Naturally fractured reservoirs have two distinct porosities, one associated with the matrix rock and another one associated with the fractures. Even if fractured reservoirs consist of irregular fractures, they can be represented by equivalent homogeneous Dual-Porosity (DP) systems [36]. The Discrete Fracture Model (DFM) represents one of the most accurate methodologies for accurately describing flow in fracture networks as it entails the direct numerical simulation of flow through the fractured porous media. Using DFM, the rock matrix and fractures are represented explicitly and Darcy flow equations are solved. However, one of the major drawbacks of this method is the high computational cost. Nevertheless, there has been an increasing interest of the scientific community as well as of the reservoir engineering community in DFM as a result also of the availability of more powerful computers and more effective discretization techniques [26, 23]. Although DFMs are becoming more and more efficient, the application of these methods at field scale is not currently realistic. Yet they are a valid tool to

perform a numerical upscaling.

On the other hand, traditional homogenisation techniques are not suitable for fractured systems because the spatial scales are not highly separated, and the topology of fractures may strongly affect the apparent (upscaled) permeability. A numerical upscaling technique based on the numerical solution of the fracture network and matrix flow at a "local" fine scale is preferable. The fluid that flows in fractures, matrix, and between matrix and fractures has different characteristics. Moreover, fracture distribution in subsurface formations usually displays significant variations in connectivity and size over the formation. Large and strongly connected fractures are typically located near bedding planes and fault zones, while small and disconnected fractures are usually located away from those regions. The variation in fracture properties, especially fracture connectivity, requires modelling different fracture zones using different numerical treatments to achieve sufficiently accurate upscaling results. Since fracture permeability is extremely high in comparison with matrix permeability, a reasonable assumption is that the flow in fractures reaches a pseudo-steady state (constant rate of change of pressure) just after the global flow starts. Transfer functions or shape factors ([35]), can thus be derived to couple the fluid flow in the matrix and fractures based on the fracture characterization, and are used to propagate the fine-scale information to the coarse-scale reservoir simulation. The assumption of instantaneous pseudo-steady state is not valid if the coarse grid block is large, or the matrix permeability is small, which is usually the case in field-scale reservoir simulations. The authors in [18, 20] introduced a systematic upscaling methodology that constructs a generalized DP model from fine-scale discrete fracture characterizations. This technique, referred to as Multiple Sub-Region (MSR) method, introduces local subregions to resolve dynamics within the matrix, and provides appropriate coarse-scale parameters that describe fracture-fracture, matrix-fracture, and matrix-matrix flow. Unlike the DP approach, in which the number of subregions inside each coarse block is reduced to one in addition to that accounting for the fractures, the MSR approach can be viewed as a generalization. In fact, several transmissibility terms are computed to represent both the interaction between fracture and matrix, and the dynamic inside the matrix, whereas, in the DP method, only one term related to fracture-fracture (block-to-block) and one term related to fracture-matrix (within block) flow is computed. The MSR approach provides a more realistic characterization of the flux exchange between matrix and fractures since the construction of the subregions reflects the actual fracture distribution. Moreover, with more than one subregion in each coarse block, MSR is capable of modeling more accurately the effects of transients inside the matrix. The upscaled properties, in particular the transmissibility, are computed by solving locally a continuity equation for the pressure, obtained by plugging Darcy's law into the mass conservation equation. Extensions of the aforementioned works are discussed in [21, 19], where additional connections between the sub-regions of the coarse cells add considered leading to a Dual-Porosity/Dual-Permeability (DPDK) model.

In this work, we have developed an accurate description of fracture networks to obtain an upscaled representation of the porous medium system through the use of a MSR technique. We have addressed the three dimensional problem directly and considered corner-point grids, typically met in reservoir simulations. The aim was to enable the computation of flow in the upscaled grid (at reservoir scale) by the computation of appropriate transmissibility between fracture and matrix, among fractures, and among different matrix blocks. This has required solving different methodological and implementation issues, which are described in the following sections. In more detail, our original contributions are: *i)* a detailed theoretical analysis of the time-dependent problem introduced in [24] to obtain the subregions, which shows that it is equivalent to a suitable stationary problem; *ii)* the numerical evidence that the steady problem introduced in [24] does actually yield different subregions in the case of non-highly connected fractures and, in this case, the time-dependent problem it is preferable; *iii)* the employment of EDFM to compute the fracture-fracture, matrix-fracture, and matrix-matrix transmissibilities in every coarse cell, with a considerable saving on the computational cost, due to the use of a Cartesian mesh built runtime. The overall procedure has been implemented in a software, in the form of a library (MSR-UpscalingLib), and an application (MSR-Upscaling) allows to run the simulations of all the test cases.

This paper starts with an introduction to the Darcy equations and to their numerical discretization through a two-point finite volume scheme in Section 2. The EDFM method is recalled in Section 3, along with the its efficient and effective implementation. Section 4 describes the upscaling procedure and the special features which we have devised, *i.e.* the multiple sub-region approach, and the treatment of the communication between two adjacent cells. A numerical assessment of the whole procedure is provided in Section 5. In particular, we address the comparison between the computational cost to build a constrained Delaunay mesh and the EDFM method for a sample of fractures, the differences for creating the sub-regions between two possible mathematical models, the assessment of the upscaling procedure by increasing the number of sub-regions, the comparison between the proposed approach and a standard software used in oil industries. Some conclusions and an outlook for future developments are gathered in Section 6.

2 Governing equations

Let $\Omega \subset \mathbb{R}^N$, for $N = 2$ or 3, be the physical domain, which represents the whole reservoir. To ease the presentation, we require that Ω be a regular domain with Lipschitz boundary, denoted by $\partial\Omega$ with unit outward normal $\boldsymbol{n}_{\partial\Omega}$. The reservoir is considered as a porous domain saturated by a liquid, *e.g.* water. We suppose that it is possible to define in Ω the representative elementary volume (REV), see, *e.g.* [4], such that the Darcy equation can be applied to describe the fluid

flow inside the domain. We are interested in computing the steady pressure field p in the whole domain Ω governed by the following Darcy system of equations

$$\begin{cases} \boldsymbol{u} + \dfrac{\Lambda}{\mu}\nabla p = \mathbf{0} & \text{in } \Omega \\ \nabla \cdot \boldsymbol{u} = q & \\ p = 0 & \text{on } \partial\Omega. \end{cases} \tag{1}$$

In the previous system, $\boldsymbol{u}$ is the Darcy velocity or the flux, q is a source/sink term, Λ is the permeability matrix, which is a symmetric and positive definite tensor, *i.e.* $\Lambda \in \mathbb{R}^{N\times N}$, and μ is the viscosity of the liquid. For simplicity, we have supposed that system (1) is supplemented with Dirichlet boundary conditions to ensure well posedeness(see, *e.g.* [8] for a detailed analysis of this problem). We have chosen a numerical scheme that requires recasting system (1) in its primal formulation, where only the pressure field p is considered as an unknown. Instead of considering system (1), we thus solve the following problem

$$\begin{cases} -\nabla \cdot \dfrac{\Lambda}{\mu}\nabla p &= q \text{ in } \Omega \\ p &= 0 \text{ on } \partial\Omega. \end{cases} \tag{2}$$

2.1 Reduced model for fractures

Several assumptions are in order. We suppose that the aperture d of the fractures is several orders of magnitude smaller than their other characteristic sizes and porous medium grid size. Moreover the permeability of the fractures may be several order of magnitude different from the permeability of the surrounding porous medium. We consider the reduced model approximation, described in [28, 3, 10], assuming that the fractures are represented by objects of co-dimension one: surfaces for $N = 3$ or lines for $N = 2$. For simplicity, we introduce the governing equations by considering a single fracture $\gamma \subset \Omega$ which cuts off entirely the reservoir and the surrounding porous medium Ω_m such that $\gamma = \Omega \setminus \overline{\Omega_m}$. The rock matrix can be split in two disjoint domains Ω_1 and Ω_2 divided by the fracture such that $\Omega_m = \Omega_1 \cup \Omega_2$ (see Figure 1 for an example. In this section,

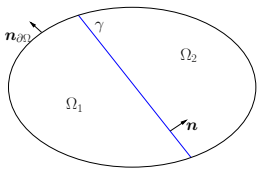

Figure 1: Domain cut by a single fracture γ approximated as a line of co-dimension one.

we indicate with a subscript 1 or 2 the restriction of the quantities defined in Ω_m to Ω_1 and Ω_2, respectively. The variables defined on the fracture γ are referred

to as "reduced" and will be indicated with a hat $\hat{\cdot}$. Moreover, we introduce the tangential operators as

$$\nabla_{\boldsymbol{\tau}} := (I - \boldsymbol{n} \otimes \boldsymbol{n}) \nabla \quad \text{and} \quad \nabla_{\boldsymbol{\tau}} \cdot := (I - \boldsymbol{n} \otimes \boldsymbol{n}) : \nabla,$$

where $\boldsymbol{n}$ is the unit outward normal to γ, pointing from Ω_1 to Ω_2. The permeability inside the fracture is assumed to be orthogonal in its intrinsic reference system, *i.e.*

$$\Lambda = \lambda_{\boldsymbol{n},\gamma} \boldsymbol{n} \otimes \boldsymbol{n} + \lambda_{\boldsymbol{\tau},\gamma} (I - \boldsymbol{n} \otimes \boldsymbol{n}) \quad \text{in} \quad \gamma,$$

where $\lambda_{\boldsymbol{n},\gamma}$ is the permeability across the fracture and $\lambda_{\boldsymbol{\tau},\gamma}$ is the permeability along the fracture. Thus, the reduced model is given by the following set of equations

$$\begin{cases} -\nabla \cdot \dfrac{\Lambda}{\mu} \nabla p = q & \text{in } \Omega_m \\ -\nabla_{\boldsymbol{\tau}} \cdot \dfrac{\hat{\lambda}}{\hat{\mu}} \nabla_{\boldsymbol{\tau}} \hat{p} = \hat{q} + 4\lambda_\gamma \left(\{\!\{p\}\!\}_\gamma - \hat{p} \right) & \text{in } \gamma, \end{cases} \tag{3a}$$

where $\hat{\lambda} := d\lambda_{\boldsymbol{\tau},\gamma}$ is the effective tangential permeability and the average operator is defined by $\{\!\{p\}\!\}_\gamma := (p_1 + p_2)/2$. The previous system is coupled with the following interface conditions between Ω_m and γ

$$\begin{cases} -\dfrac{\Lambda}{\mu} \nabla p_1 \cdot \boldsymbol{n} = 2\lambda_\gamma (p_1 - \hat{p}) \\ -\dfrac{\Lambda}{\mu} \nabla p_2 \cdot \boldsymbol{n} = 2\lambda_\gamma (\hat{p} - p_2) \end{cases} \quad \text{on } \gamma, \tag{3b}$$

with $\lambda_\gamma := \lambda_{\boldsymbol{n},\gamma}/d$ the effective normal permeability. To generalize system (3) to the case of several fractures, we suppose that the reservoir Ω can be split in two disjoint subsets: $\Omega_f \subset \Omega$, which represents the fractures set of the reservoir, and $\Omega_m \subset \Omega$, which represents the rock matrix of the reservoir. Thanks to the previous assumptions, domain Ω_f consists of the union of N_f, possibly intersecting, objects of co-dimension one, each indicated by γ_i, such that $\Omega_f = \cup_{i=1}^{N_f} \gamma_i$. We indicate by a subscript the quantities introduced previously for a single fracture. Given a fracture γ_i it is still possible to define the two portions of Ω_m defined on the same and on the opposite sides of the fracture normal to γ_i. To couple the problems in Ω_m and in Ω_f, we consider the following interface conditions (see [2, 1] for instance), between intersecting fractures

$$\begin{cases} \hat{p}_i = \hat{p}_j \\ -\dfrac{\hat{\lambda}_i}{\hat{\mu}} \nabla_{\boldsymbol{\tau}_i} \hat{p}_i \cdot \boldsymbol{\tau}_i = -\dfrac{\hat{\lambda}_j}{\hat{\mu}} \nabla_{\boldsymbol{\tau}_j} \hat{p}_j \cdot \boldsymbol{\tau}_j \end{cases} \quad \text{on } \gamma_i, \gamma_j \text{ for } i \neq j, \tag{4a}$$

and the condition at each internal tip of the fractures

$$-\frac{\hat{\lambda}_i}{\hat{\mu}}\nabla_{\boldsymbol{\tau}_i}\hat{p}_i\cdot\boldsymbol{\tau}_i=0 \quad \text{for} \quad \partial\gamma_i\setminus\partial\Omega. \tag{4b}$$

For a more general type of coupling conditions between intersecting fractures, we refer to [16, 17], while for a detailed description of the condition on the fractures tips, see *e.g.* [3]. Problem (3), written for each fracture γ_i, coupled with (4) represents the global set of equations that describe the pressure field inside the whole reservoir Ω.

Remark 1. *Even though, in this work, the permeability of each fracture takes values which much larger than the permeability of the rock matrix, the proposed model can also handle fractures that behave either like barriers or as a preferential path for the flow, or even more complex situations. However, in our case, the porous media flow tends to focus along the fractures and inside the fractures networks (see [6, 5] for an approximation where only the fractures networks are considered).*

2.2 Numerical approximation

To solve the problem numerically, we resort to the classical two-point finite volume scheme, see [13] for a detailed description. This is justified by its local mass conservation property, and by the resulting M-matrix property of the stiffness matrix. Moreover, it can be easily used to solve single as well as multi-phase flows, and it is one of the fastest scheme available in the literature for solving diffusive partial differential equations. We introduce the scheme for a general grid cell, then the application to a particular problem is straightforward. For each cell K of the computational grid, the scheme can be written as

$$\sum_{\sigma\in\mathcal{E}_K} F_{\sigma,K}(p)=\int_K q\mathrm{d}\boldsymbol{x},$$

where $\mathcal{E}_K$ is the set of facets (edges in $2D$ and faces in $3D$) σ of the cell K, and $F_{\sigma,K}$ is the two-point flux approximation of the diffusive operator across the facet σ. Given two adjacent cells, K and L, sharing the facet σ, we enforce the local conservation of mass through σ: $F_{\sigma,K}(p)+F_{\sigma,L}(p)=0$. The two-point flux scheme approximate the flux in the following way:

$$F_\sigma(p):=\frac{T_{KL}}{\mu}\left(p_K-p_L\right), \tag{5}$$

where p_K and p_L are the pressures in the cells K and L, respectively. Since $T_{KL}=T_{LK}$, with this definition, the conservation of mass is automatically satisfied. In (5), we have introduced the transmissibility T_{KL} between adjacent cells, which is the main ingredient of the two-point flux approximation scheme.

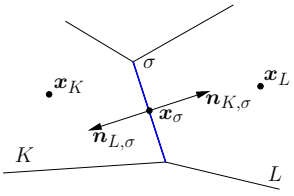

Figure 2: Some notation used to define F_σ and T_{KL}.

The computation of the transmissibility is based on some geometrical quantities of the two cells K and L, as well as on the permeability tensor. The transmissibility can be computed through the following formula:

$$T_{KL} = \frac{T_K T_L}{T_K + T_L}, \tag{6}$$

where T_K and T_L are the so called "half transmissibilities" related to the cells K and L, respectively. These objects, referring to the notation in Figure 2, can be computed by

$$T_K := \frac{(\boldsymbol{x}_K - \boldsymbol{x}_\sigma)^\top}{\|\boldsymbol{x}_K - \boldsymbol{x}_\sigma\|_2^2} \Lambda \boldsymbol{n}_{K,\sigma} \, |\sigma| \,, \tag{7}$$

with $\boldsymbol{n}_{K,\sigma}$ the unit outward normal to $\sigma \in \mathcal{E}_K$, $\boldsymbol{x}_K$ the centre of mass of the cell K, and $\boldsymbol{x}_\sigma$ the centroid of the facet σ.

2.3 Virtual fracture cells

Another possible approach, commonly used in the approximation via DFM, is based on the construction of an additional mesh consisting of *virtual fracture cells* to discretize the problem in each fracture (see [15] for a detailed presentation and analysis of this approach). The advantage of this method is to avoid the construction of the tangential operators and the imposition of the coupling conditions of problem (3) approximating equation (2) in the whole domain Ω. The virtual fracture cells are obtained by extrusion of fracture facets along their normal direction by a thickness $\pm d/2$. The global mesh is obtained by the union of the mesh which discretizes the rock matrix and the mesh comprising the virtual fracture cells (see Figure 3 as an example of the construction of virtual fracture cells).

Remark 2. *In practice, the virtual fracture cells are never constructed explicitly, but only introduced to compute the geometric quantities required by the two-point approximation.*

Remark 3. *In Appendix A, we show the equivalence, when employing the two-point scheme, between the discrete version of* (3) *and* (2) *discretized using the virtual fracture cells. Therefore, the implementation of the two approaches does provide the same results.*

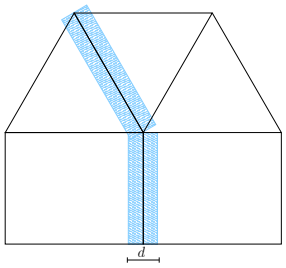

Figure 3: Example of the construction of virtual fracture cells.

3 Embedded discrete fracture model

To solve problem (3) as well as (2) with a two-point finite volume scheme, a discretization of both porous medium and fractures is required. We focus our attention on complex three-dimensional problems. A representation of the computational grid with a conforming mesh that adapts to the fractures may be very complex to generate, due to the small angles that may occur at the intersection of two or more fractures. Moreover, this procedure could generate many very small elements that could badly affect the numerical solution. Finally, since the typical grids used in the commercial reservoir softwares are the so-called corner-point grids, it is convenient to employ such kind of grid in order to easily interface with these programs.

In order to avoid having to build a conforming mesh and to alleviate the computational cost, we rely on the methodology first introduced by [26] and expanded in [32, 31]. This method, known has Embedded Discrete Fracture Model (EDFM), allows one to generate the fractures mesh and porous medium grid independently, thus avoiding the need for a conforming mesh and allowing one to use corner-point grids. Fractures, which are approximated with planar quadri-

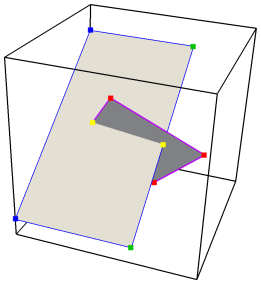

Figure 4: Example of fractures mesh generated with EDFM. It is required only to compute the intersections between the fractures and between each fracture with the coarse cell.

laterals, are meshed so that each corner-point cell that they intersect contains one and only one fracture cell per fracture. Intersections between fractures are

computed, but do not affect the meshing strategy of either fractures or corner-point grid. They are instead used to compute transmissibilities between different fracture cells (see Figure 4 for an example).

To each matrix cell and to each fracture cell a degree of freedom is assigned. This means that transmissibilities between matrix and fracture cells, as well as those between different fracture cells need be computed.

The first step towards computing all transmissibilities is to actually mesh the fractures. This means individuating the corner point cells intersected by each fracture, and computing the polygons resulting from the intersection of the fracture with each cell. The vertices V of each polygon are the union of three

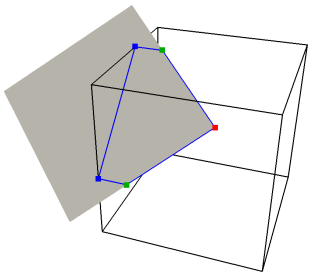

Figure 5: The three types of points found by intersecting a fracture, in grey, with a corner-point cell: V_i in red, V_f in green, and V_e in blue. The resulting fracture cell is the one with the blue edges.

sets of points: fracture vertices lying inside the element V_i, intersections between an element face and a fracture edge V_f, and intersections between an element edge and the fracture V_e

$$V = V_i \cup V_f \cup V_e,$$

(see Figure 5 for an example). The second and third set of points, V_f and V_e, can be found by computing the intersections between a line segment, the fracture edge or the element edge, respectively, and a bilinear surface, the element face or the fracture. Given the parametrization of a bilinear surface of vertices $\boldsymbol{a}$, $\boldsymbol{b}$, $\boldsymbol{c}$, and $\boldsymbol{d}$

$$\boldsymbol{r}(u, v) = v(u\boldsymbol{b} + (1-u)\boldsymbol{a}) + (1-v)(u\boldsymbol{c} + (1-u)\boldsymbol{d}),$$

finding the intersections with a line segment

$$\boldsymbol{s}(t) = t\boldsymbol{p} + (1-t)\boldsymbol{q},$$

with endpoints $\boldsymbol{p}$ and $\boldsymbol{q}$, means solving the non-linear system

$$\boldsymbol{r}(u, v) = \boldsymbol{s}(t),$$

which can be solved explicitly avoiding the need for iterative methods. Lastly, testing whether a fracture vertex lies inside a trilinear element, *i.e.* finding V_i,

can be done using a ray-tracing algorithm. In particular, we can build a line segment going from the element parametric centre to the point whose position we want to check. We can then count the total number of intersections between this segment and each element face, again, by finding the intersection between a line and a bilinear surface. An odd number of intersections means that the point is outside the element, an even number means that it is inside, as stated by the ray casting algorithm.

To reduce the number of corner-point cells to be tested for intersections with a specific fracture, a search structure on the corner-point grid can be used. For this purpose, we chose an ADTree, which, given the bounding box of a fracture, provides a possible list of cells that it intersects.

The intersections between different fracture cells, required to compute transmissibilities, are instead found by intersecting the edges of one fracture with the surface of the other. Since we already know in which corner-point cell each fracture cell falls, we can restrict the computation of intersections to those in the same corner-point cell, thus greatly reducing the computational cost.

Remark 4. *Because corner point cells are represented as trilinear elements, their faces are bilinear surfaces. Actually, their intersections with a fracture are therefore not linear. In order to avoid having to deal with complex geometric objects and to be able to use simple closed-formed formulas to compute transmissibilities, these intersections are, however, approximated with line segments.*

Once we have defined all fracture cells and found the intersections between them, we can proceed to compute the transmissibilities. Let us denote with A_Σ the surface area of the fracture cell Σ contained in a matrix block K, with d_i its aperture and with $\boldsymbol{n}_\Sigma$ the unit normal vector to the fracture surface. The transmissibility between a matrix cell and a fracture cell contained in it is computed using the formula proposed in [26]:

$$T_{K\Sigma} = A\frac{\boldsymbol{n}_\Sigma^\top \Lambda \cdot \boldsymbol{n}_\Sigma}{d_{K\Sigma}},$$

where $d_{K\Sigma}$ denotes the average distance between a generic point of the fracture cell and the fracture plane. Transmissibility between two fracture cells, Σ and Θ, is instead computed similarly to (6):

$$T_{\Sigma\Theta} = \frac{T_\Sigma T_\Theta}{T_\Sigma + T_\Theta}. \tag{8}$$

We consider now two adjacent fracture cells sharing the interface σ. However, σ is the representation of two distinct facesi, σ_Θ and σ_Σ, with their own normals $\boldsymbol{n}_{\sigma_\Theta}$ and $\boldsymbol{n}_{\sigma_\Sigma}$, respectively. The half transmissibilities, T_Σ and T_Θ, are computed locally and are given by the formula derived from the two-point flux approximation found in [23]

$$T_\Theta = \frac{|\sigma_\Theta|\, \lambda_i d_i \left(\boldsymbol{x}_{\sigma_\Theta} - \boldsymbol{x}_\Theta\right) \cdot \boldsymbol{n}_{\sigma_\Theta}}{d_{\Theta\sigma_\Theta} \|\boldsymbol{x}_{\sigma_\Theta} - \boldsymbol{x}_\Theta\|_2}.$$

A similar formula can be provided for T_Σ. In the previous formula λ_i, d_i and $\boldsymbol{x}_i$ are the permeability, aperture and centre of mass, respectivedly, of fracture cell Θ, $\boldsymbol{x}_{\sigma_\Theta}$ is the centroid of σ_Θ, and $d_{\Theta\sigma_\Theta}$ is the average distance between a generic point of Θ and σ_Θ, computed by

$$d_{\Theta\sigma_\Theta} = \frac{1}{|\Theta|}\int_\Theta \|\boldsymbol{x}_{\sigma_\Theta} - \boldsymbol{x}\|_2 \mathrm{d}\boldsymbol{x}.$$

A similar formula can be used for Σ. For fracture cells Σ_i belonging to different fractures γ_i and intersecting each other on line segment s, the half transmissibility is instead computed as

$$T_{\Sigma_i} = \frac{|s|\,\lambda_i d_i}{d_{\Sigma_i s}},$$

where $d_{\Sigma_i s}$ is the average distance of a generic point of Σ_i to the line to which s belongs. EDFM has multiple advantages that come in handy to make the

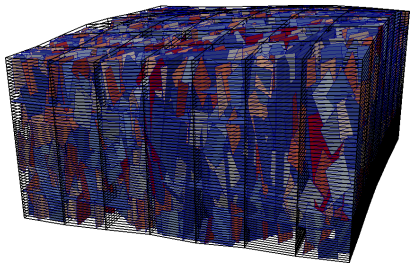

Figure 6: Example of complex fracture configuration meshed with EDFM: 4670 fractures immersed in a $7 \times 7 \times 60$ corner-point grid.

MSR upscaling methodology as robust and fast as possible. On the one hand, its ability to handle complex fracture geometries (see Figure 6 as an example) allows us to handle virtually any configuration. On the other hand, its speed in generating meshes and computing transmissibilities makes it particularly suitable to be used in a scenario like that of Multiple Sub-Regions upscaling, where a multitude of problems need be solved, and where it can also help in identifying fracture networks, *i.e.* all those fracture cells that are somehow connected one to another.

4 Upscaling procedure

In the case of real applications, where thousands of fractures are considered, it is very costly to construct a conforming mesh with a good quality to approximate and solve problem (2) with the virtual fracture cells. Moreover, due to the uncertainty in the underground, it can be necessary to perform multiple scenario analyse by changing the fractures position, thus making unreasonable and

unaffordable building a different mesh for each realization. For these reasons, we have chosen a different strategy to solve the global problem, considering an upscaling technique to derive effective model parameters, at a coarser scale level, which accurately represent the fine scale model. An exhaustive presentation of the classical upscaling techniques, both numerical and analytical, in unfractured porous media, can be found in [11]. Nevertheless, in this work we focus our attention on fractured reservoirs. The main idea of the numerical upscaling is to consider a global coarse grid $\mathcal{G}$ which describes the upscaled reservoir and that it is geometrically independent of the fractures Ω_f. Typically, $\mathcal{G}$ is a corner-point grid, which represents the different sedimentary layers of the reservoir (see Figure 7 as an example of $\mathcal{G}$). The upscaling procedure computes the following upscaled

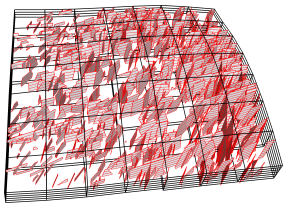

Figure 7: Example of a coarse grid in a fractured reservoir.

proprieties: the transmissibility between adjacent coarse blocks, the depth, and the porous volume of each coarse block. However, the methods described in [11], as many other techniques proposed in the literature, do not explicitly consider the presence of the fractures. For this reason, we have considered the upscaling procedure proposed in [24, 18].

To ease the presentation we recall the basic ideas of the upscaling scheme, extended by adding a matrix-matrix connection between coarse cells.

4.1 Multiple sub-regions

In this section, we present the upscaling procedure proposed by [24]. Considering two adjacent coarse cells of the global mesh $\mathcal{G}$, the main assumption of the scheme is that the flow across the two cells takes place only through the fractures networks and not through the porous medium. This hypotheses is motivated by the fracture permeability, which is several order of magnitude greater than the permeability of the rock matrix. Moreover, we assume that the interchange of flow between the fractures and the rock matrix is localized only inside each coarse cell. The method of multiple sub-regions, introduced in the aforementioned work, considers a sub-division of each coarse cell to enhance the description of the porous medium and to obtain better results. The resulting degrees of freedom (*d.o.f.*s) of the coarse model are associated with each coarse cell and represent the pressure inside the local fractures networks and the pressure in each sub-region of the rock matrix. Following the idea proposed in [24], the upscaled model could be sketched with a graph, where each node represents a degree

of freedom, and each link represents the connection, *i.e.* , the transmissbility, between the degrees of freedom (see Figure 8). In the sequel, to compute the upscaled quantities we solve local problems involving single cells, or pairs of cells. Considering a general coarse block $K \in \mathcal{G}$, we need to build a finer grid, indicated by $\mathcal{M}_K$, composed by C_K polyhedra in $3D$, polygons in $2D$, or briefly cells, indicated by k, such that

$$\overline{K} = \bigcup_{i=1}^{C_K} \overline{k_i},$$

which is used to solve the local problems. In the case when two adjacent coarse blocks $K, L \in \mathcal{G}$ are involved in the computation, we construct the fine mesh, indicated by $\mathcal{M}_{KL}$, as the union of each sub-mesh $\mathcal{M}_{KL} = \mathcal{M}_K \cup \mathcal{M}_L$. The interface Γ_{KL} between K and L is then honoured by the discretization and, for simplicity, we require that the resulting mesh be conforming. In each pair of coarse cells it is possible to identify a set of local networks of fractures with N_{KL} elements. We indicate by $\mathcal{N}_{KL}^i$ the mesh associated with the discretization of the i-th local network of fractures inside the pair of cells K and L. The global set of fractures for pair of cells is denoted by $\mathcal{N}_{KL}$. We have

$$\mathcal{N}_{KL} = \bigcup_{i=1}^{N_{KL}} \mathcal{N}_{KL}^i \quad \text{and} \quad \mathcal{N}_{KL}^i \cap \mathcal{N}_{KL}^j = \emptyset \text{ for } i \neq j.$$

We indicate also by $\mathcal{N}_K^i$ and $\mathcal{N}_K$ the restriction of $\mathcal{N}_{KL}^i$ and $\mathcal{N}_{KL}$ to the single coarse cell K, respectively. We indicate by $\mathcal{E}_A$ the set of edges of the mesh A, indicated in the subscript. Finally, given an edge σ, we denote by $\mathcal{M}_\sigma$ the set of cells having σ as an edge.

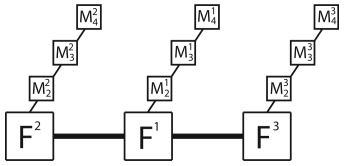

Figure 8: The graph represents three coarse cells in which each of them contains a system of fractures and three porous matrix sub-regions. The communication between the coarse cells occurs only through the fractures.

4.1.1 Inter-cell upscaled transmissibility

To compute the upscaled transmissibility between two coarse blocks $K, L \in \mathcal{G}$, which represents the link between the coarse degrees of freedom associated with the local fractures sets of the two cells, we solve a diffusion problem where a pressure gradient is imposed across the boundary of the cell. In practice, we

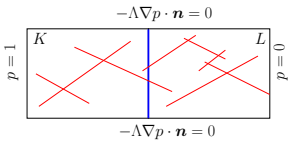

Figure 9: Schematic example of the local problem used to compute the upscaled fracture-fracture transmissibility through the interface of the coarse cells. The interface between the coarse cells is depicted in blue.

solve the system of equations (2), by the virtual fracture cells approach, with boundary conditions defined as in Figure 9. The upscaling procedure is based on the formula (5), where averaged and global quantities are considered. Once the numerical solution is computed, it is possible to calculate the average pressure on each coarse cell and the flux across the coarse cells interface. The number of local networks of fractures which actually cross the interface Γ_{KL} is indicated by N^{Γ}_{KL}. We sort the local networks of fractures such that, for a given $j = 1, \dots, N^{\Gamma}_{KL}$, we have $\mathcal{N}^j_{KL} = \mathcal{N}^j_K \cup \mathcal{N}^j_L$. For each local network of fractures of the coarse cell K the averaged pressure $\overline{p}^j_K$ is computed by

$$\overline{p}^j_K = \frac{\sum_{k \in \mathcal{N}^j_K} |k|\, p_k}{\sum_{k \in \mathcal{N}^j_K} |k|} \quad \text{for } j = 1, \dots, N^{\Gamma}_{KL},$$

and analogously fo L. It is important to remark that only the fracture cells are involved in the computation of $\overline{p}^j_K$ and $\overline{p}^j_L$. The total flux, indicated by F^j_{KL}, through each fractures network $\mathcal{N}^j_{KL}$ across the interface between the two cells is computed by

$$F^j_{KL}(p) = \sum_{\sigma \in \mathcal{E}^j_{\Gamma}} T_{kl}\,(p_k - p_l) \quad \text{for } j = 1, \dots, N^{\Gamma}_{KL},$$

where $\mathcal{E}^j_{\Gamma} := \mathcal{E}_{\mathcal{N}^j_K} \cap \mathcal{E}_{\mathcal{N}^j_L}$ is the set of the interface edges. In the previous formula, we have assumed that, given $\mathcal{N}^j_{KL} = \mathcal{N}^j_K \cup \mathcal{N}^j_L$, every fine cell $k \in \mathcal{M}_\sigma$ belongs to $\mathcal{N}^j_K$ and every cell $l \in \mathcal{M}_\sigma$ belongs to L. Finally, the upscaled transmissibility between K and L can be computed considering

$$T_{KL} = \mu \sum_{j=1}^{N^{\Gamma}_{KL}} \left| \frac{F^j_{KL}(p)}{\overline{p}^j_K - \overline{p}^j_L} \right|.$$

This case is represented by the graph in Figure 10.

4.1.2 Intra-cell uspcaled transmissibility

Once the upscaled transmissibility is computed for each pair of cells in the coarse grid, the multiple sub-regions subdivides each single coarse cell $K \in \mathcal{G}$ into

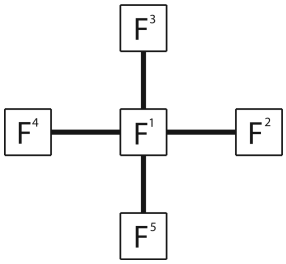

Figure 10: Graph that represents the connections between the coarse cells. Each node represents the system of fracture inside the coarse cell, while each link between the nodes represents the connection, *i.e.* the transmissibility.

several parts and computes the upscaled transmissibility between the degrees of freedom associated with the fractures set of the cell and the first sub-region, as well as the transmissibilities between a sub-region and the subsequent one. To divide the coarse cell, the authors of [24] propose two different strategies, claiming that they provide equivalent results. However, this is true only if the fractures set, local to the coarse cell, is connected (see the numerical experiments reported in Subsection 5.2). The first strategy considers the compressible single-phase flow equation for the generic coarse cell $K \in \mathcal{G}$

$$\begin{cases} \phi c \dfrac{\partial p}{\partial t} - \nabla \cdot \dfrac{\Lambda}{\mu} \nabla p = q_f & \text{in } K \text{ for } t > 0 \\ -\dfrac{\Lambda}{\mu} \nabla p \cdot \boldsymbol{n}_K = 0 & \text{on } \partial K \text{ for } t > 0 \\ p = 0 & \text{in } K \text{ for } t = 0, \end{cases} \tag{9}$$

where ϕ is the porosity, c the compressibility and q_f is a piecewise constant source term strictly positive inside the fractures and zero elsewhere (see Figure 11 for a representation of the problem). This equation is solved until a pseudo-steady state is reached, *i.e.* until the following condition is fulfilled

$$\frac{\partial p}{\partial t} = c_1 = const.$$

Finding the pseudo-steady state of problem (9) is equivalent to finding the solution of a suitable stationary problem, as detailed in Appendix B. Moreover, problem (9), is weakly coercive and the numerical errors accumulated during the temporal loop may alter the computed solution. In the following tests, we will consider only the stationary formulation of problem (9). To numerically solve problem (9), we consider again a fine mesh $\mathcal{M}_K$ built from K, and we use the two-point flux approximation to compute the numerical solution. We consider S_K sub-regions for the coarse cell $K \in \mathcal{G}$. The last sub-region is related to the fractures set. The other $S_K - 1$ are defined through the iso-pressure contours on the fine grid $\mathcal{M}_K$. Thanks to the characteristics of problem (9), each sub-region

is linked between the previous and the subsequent one. Once the sub-regions are defined, it is possible to compute the transmissibility between each pair of sub-regions considering a proper generalization of formula (5). We indicate by $\mathcal{M}_K^i$ the cells of the fine mesh $\mathcal{M}_K$ related to the sub-region i and by $\mathcal{E}_K^i$ the edges related to $\mathcal{M}_K^i$. We consider also the set of edges between two sub-regions as $\mathcal{E}_K^{i,i+1} = \mathcal{E}_K^i \cap \mathcal{E}_K^{i+1}$. Finally, we set $\mathcal{M}_K^{S_K} = \mathcal{N}_K$ and $\mathcal{E}_K^{S_K-1,S_K} = \mathcal{N}_K$. We compute the mean pressure for each sub-region i as

$$\overline{p}_K^i = \frac{\sum_{k\in\mathcal{M}_K^i} |k|\, p_k}{\sum_{k\in\mathcal{M}_K^i} |k|} \quad \text{for } i = 1, \dots, S_K,$$

as well as the flux between the sub-regions i and $i+1$ as

$$F_K^{i,i+1}(p) = \sum_{\sigma\in\mathcal{E}_K^{i,i+1}} T_{kl}\,(p_k - p_l) \quad \text{for } i = 1, \dots, S_K - 1,$$

where we have assumed that every fine cell $k \in \mathcal{M}_K^i$ and $l \in \mathcal{M}_K^{i+1}$. The upscaled transmissibiliy between the sub-regions i and $i+1$ is computed by

$$T_K^{i,i+1} = \mu \left| \frac{F_K^{i,i+1}(p)}{\overline{p}_K^i - \overline{p}_K^{i+1}} \right|.$$

Following [24], to define the sub-regions of a coarse cell $K \in \mathcal{G}$, it is possible to consider instead of problem (9) another problem that we refer to as "fixed-pressure" approach. We consider the pressure field, numerical solution to the following stationary problem

$$\begin{cases} -\nabla \cdot \dfrac{\Lambda}{\mu} \nabla p = q & \text{in } K \setminus \mathcal{N}_K \\ p = p_0 & \text{in } \mathcal{N}_K \\ -\dfrac{\Lambda}{\mu} \nabla p \cdot \boldsymbol{n}_K = 0 & \text{on } \partial K \setminus \mathcal{N}_K, \end{cases} \tag{10}$$

where the source term is constant everywhere in the porous matrix and is zero inside the fractures, whereas in the fractures the solution is fixed and constant everywhere. To ease implementation, it is possible to choose $p_0 = 0$ and $q < 0$. Figure 12 represents by a graph the case of a single coarse cell with the fractures and the subregions. Once the sub-regions are defined, the upscaled transmissibilities among them are computed as in the previous strategy.

In the aforementioned work, the authors claim that the two approaches give qualitatively equivalent results, however we will show in Section 5.2 that this is not always true in the case of disconnected networks of fractures.

Remark 5. *The fractures that are fully-immersed, i.e. , the fractures that are contained only in one coarse cell, are considered as part of the sub-region in which they belong to.*

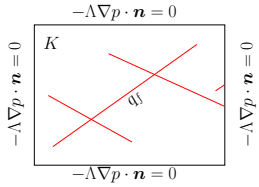

Figure 11: Schematic example of the local problem used to compute the upscaled fracture-matrix and matrix-matrix transmissibilities.

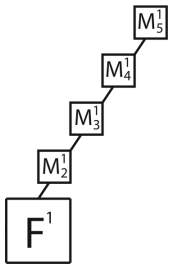

Figure 12: Graph that represents the connections between the system of fractures and the sub-regions of a single coarse cell. The link between the node related to the fractures and the first sub-region represents the fracture-matrix transmissibility, while the remaining links represent the transmissibility between subsequent sub-regions.

Remark 6. *The assumption that each sub-region is linked between its previous and subsequent sub-region does not apply in the pseudo-steady approach (or its equivalent steady-state formulation). By considering the "fixed-pressure" approach, the same effect may occur when the local mesh is not fine enough. In these cases, we allow that each sub-region can be linked to all the others sub-regions.*

4.2 Matrix-matrix comunication

In this section, we present a new algorithm to deal with disconnected networks of fractures in the upscaling procedure. Disconnected networks of fractures occur when there are pairs of coarse cells where the local set of fractures does not intersect the interface between the coarse cells. In this case, the previous algorithm does not compute any upscaled transmissibility since one of the hypotheses of the standard multiple sub-regions method is that the flow, between adjacent coarse cells, takes place only through the fractures. Figure 13 shows this kind of configurations. In particular, we subdivide them into three cases: both cells do not contain any fracture; only one cell contains the fractures; both cells contain fractures but they do not cut the common interface. Besides these cases, for a more accurate representation, we extend the procedure described

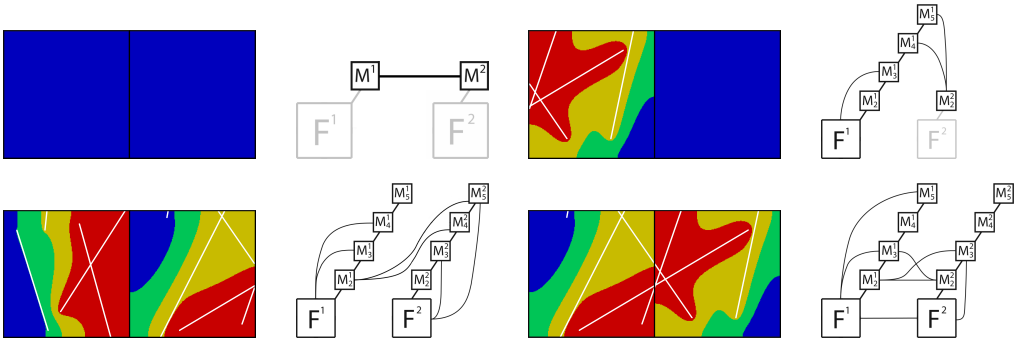

Figure 13: Examples of the four possible configurations of pair of coarse cells in which we consider the matrix-matrix connection.

below also in the case in which the common interface is cut by the fractures.

To handle these cases, we take advantage of the created sub-regions during the computation of the upscaled transmissibilities inside each coarse cell. Given a pair of adjacent coarse cells, where their networks of fractures are disconnected, we solve the diffusion problem (2), where a pressure gradient is imposed across their boundary, as depicted in Figure 9. Once the solution is obtained, we compute the transmissibilities between each sub-region of the first coarse cell which faces, through the interface between the coarse cells, any sub-regions of the other coarse cell. An example is given in Figure 13 (bottom left), where the sub-regions of a pair of coarse cells are represented. The graph depicts the logical connections between the *d.o.f.*s associated with the sub-regions: the sub-regions of the left cell (indicated with 1) are linked to the sub-regions of the right cell (indicated with 2). For each link we have a transmissibility value. Considering the two fine meshes $\mathcal{M}_K$ and $\mathcal{M}_L$ associated with the pair of coarse cells K and $L \in \mathcal{M}$, it is possible to define the sub-regions as explained in the previous section obtaining S_K and S_L sub-regions for the two coarse cells K and L, respectively. As in the previous section, the upscaled transmissibility is based on a proper generalization of equation (5). We compute the mean pressure for each sub-region i of the coarse cell K as

$$\overline{p}_K^i = \frac{\sum_{k \in \mathcal{M}_K^i} |k| \, p_k}{\sum_{k \in \mathcal{M}_K^i} |k|},$$

and analogously for the coarse cell L. We indicate the set of edges of the sub-region i of the coarse cell K which faces the sub-region s of the coarse cell L as $\mathcal{E}_{KL}^{i,s}$. The total flux, indicated by $F_{KL}^{i,s}$, through the interface between the cells of the sub-region i, referred to the coarse cell K, and the sub-region s, referred to the coarse cell L, is computed by

$$F_{KL}^{i,s}(p) = \sum_{\sigma \in \mathcal{E}_{KL}^{i,s}} T_{kl} \left(p_k - p_l \right),$$

where every fine cell $k \in \mathcal{M}_\sigma$ belongs to the sub-region i of the coarse cell K, and every fine cell $l \in \mathcal{M}_\sigma$ belongs to the sub-region s of the coarse cell L. Finally, the upscaled transmissibility between the sub-region i of the coarse cell K and the sub-region s of the coarse cell L can be computed considering

$$T_{KL}^{i,s} = \mu \left| \frac{F_{KL}^{i,s}(p)}{\overline{p}_K^i - \overline{p}_L^s} \right|.$$

It is important to notice that the procedure does not increase the number of degrees of freedom but only their connections. In the particular case when one of the two coarse cells does not contain fractures, as represented in Figure 13 (top right), we assume that the latter has only one sub-region and the upscaling algorithm remains the same. All the sub-regions, facing the interface between the coarse cells of the left coarse cell are connected with the unique sub-region of the right coarse cell. This procedure is particular important in the "boundary" of a fractures network to allow the spreading of the flow in the neighbouring coarse cells, and to link different networks of fractures.

To complete the upscaling procedure for a general reservoir, as in Figure 7, it is possible that some pairs of coarse cells do not contain any fracture. In this case, we consider the standard formula (6) and (7), with the geometric properties related to the coarse mesh $\mathcal{M}$. This case is represented by the graph shown in Figure Figure 13 (top left). The degrees of freedom related to the fractures are not considered, and for each coarse cell there is a single *d.o.f.* associated with the unique matrix sub-region, *i.e.* , the entire coarse cell. Hence, the link between the coarse cells is due to the flow through the porous matrix.

Finally, to improve the accuracy of the procedure, we compute the matrix-matrix transmissibility also in the case of pair of cells in which the common interface is fractured. Indeed, though most of the flow go through the fractures, the interface area associated with the porous matrix may be very large with respect to the one related to the fractures, so that the flow transmitted by the matrix may be significant. An example of this case is shown in Figure 13 (bottom right): besides the fracture-fracture trasmissibility, a communication between the sub-regions of the two coarse cells is taken into account.

Remark 7. *To speed up the resolution in the implementation of the algorithm it is possible to avoid the computation of the transmissibilities among the sub-regions if the local set of fractures belongs to the same global network of fractures. This hypotheses is reasonable since the transmissibility of the fractures is several orders of magnitude greater than the transmissibility of the rock matrix.*

5 Numerical examples

In this section, we present several numerical examples to assess the performance of the method presented in the previous sections.

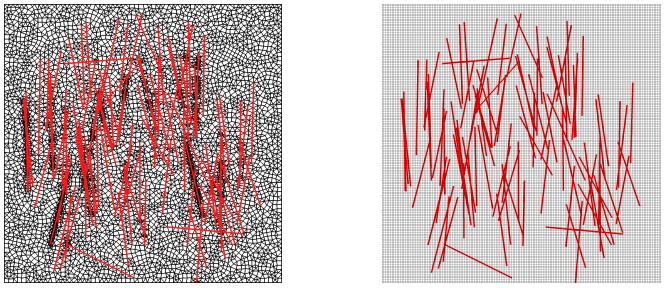

Figure 14: Left: mesh built with a constrained Delaunay algorithm. Right: Cartesian grid used for EDFM.

5.1 Computational efficiency of EDFM

We present a comparison of the computational cost between EDFM, introduced in Section 3, and a classical constrained Delaunay approach with a procedure to agglomerate triangular elements into quadrilaterals. In particular, we assess the computational time for the construction of the grid and the assembly of the stiffness matrix for a local problem. In this test we consider only two-dimensional meshes. Since the domain is a square, to construct the grid for EFDM we have considered a Cartesian grid with the same number of rock matrix cells as the Delaunay grid. We notice from Figure 14 that the grid created by the constrained Delaunay algorithm is matching with all the fractures and has many elements near the fractures. The rock matrix cells are 137780 and 136160 for the EDFM and Delaunay grids, respectively. The fracture faces are 27110 and 45225 for the EDFM and Delaunay grids, respectively. The computational cost for the mesh construction is $1.577s$ for EDFM and $427s$ for constrained Delaunay, while the cost of assembly stiffness matrix is $0.202s$ for EDFM and $0.438s$ for constrained Delaunay. It is evident that the main computational cost in the simulation is due to the mesh generation. The cost for the mesh construction in the case of EDFM includes the generation of the Cartesian grid and the algorithm which computes all the intersections between the fractures and the rock matrix, and among the fractures. The cost to assembly the stiffness matrix is comparable in both approaches. We highlight that in the case of realistic and three-dimensional domains, the discrepancy between the approaches is more evident.

Remark 8. *A more complete comparison, to verify the efficiency of EDFM among other grid generation techniques, has to include also the simplification techniques for complex networks of fractures. See for example [27, 29].*

Remark 9. *In all the following test cases, the fine grid for each coarse cell is constructed via a tensor product of the one-dimensional grids of their edges. See Figure 15 for some examples.*

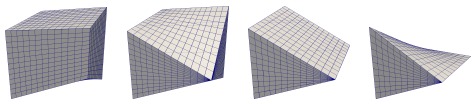

Figure 15: Examples of fine grids from single coarse cells. From the left: general hexahedron with bilinear surfaces and degenerate hexahedron with one, two, and three pillars pinched. The case of two opposite pinched pillars is not considered.

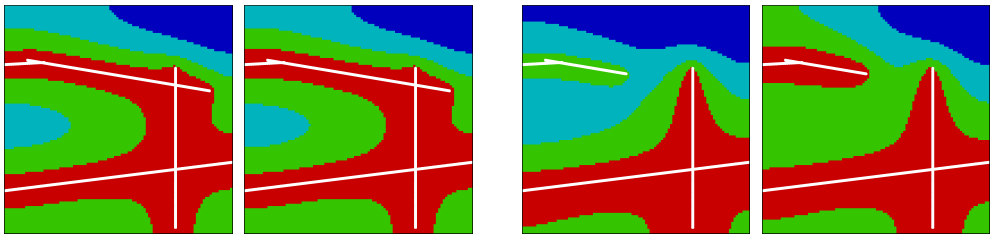

Figure 16: Representation of the sub-regions when the pseudo-steady problem is considered or when the pressure is imposed inside the fractures. In the left block the fractures are more connected than in the right block.

5.2 Comparison of single-cell solution

We compare the two strategies to compute the sub-regions given by the problems (9) and (10). In the following tests we assume four sub-regions for the porous matrix.

Connected networks of fractures We consider a coarse cell where all the fractures are connected. The computed sub-regions of the two approaches are represented in the left part of Figure 16. We clearly see that, in this case, the computed sub-regions are qualitatively equivalent. In Table 1 we report the values of the computed upscaled transmissibilities. We notice that the values are similar for the two strategies.

Disconnected networks of fractures We consider a second test case where the fractures are less connected than the previous test case. The computed sub-regions of the two approaches are represented in the right part of Figure 16.

	Problem 9	Problem 10
fractures-red	123.47	122.18
red-green	11.94	11.85
green-cyan	12.37	12.34
cyan-blue	4.32	4.3

Table 1: Comparison of the upscaled transmissibilities among the sub-regions defined in the left part of Figure 16.

	Problem 9	Problem 10
fractures-red	190	92.24
fractures-green	9.62	-
red-green	6.87	10.20
green-cyan	11.88	8.76
cyan-blue	6.61	4.05

Table 2: Comparison of the upscaled transmissibilities among the sub-regions defined in the right part of Figure 16.

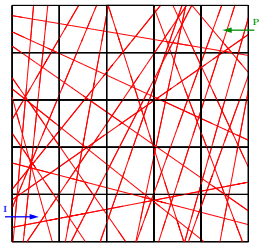

Figure 17: Coarse mesh and fractures used to assess the multiple sub-regions technique.

We see that, in this case, the sub-regions are very different and the sub-region related to the fractures is connected with two different sub-regions. In Table 2 we report the values of the computed upscaled transmissibilities. We notice that the values are completely different for the two strategies.

In conclusion, problem (9) provides a different weight to each fracture network inside the cell, whereas problem (10) treats all the fractures with the same weight. In the particular case in which the coarse cell is fully connected in a unique network, the two problems behave in the same way.

5.3 Assessment of the upscaling procedure

In this part we test the effect of increasing the number of porous matrix sub-regions against a reference solution, to verify a better approximation when the number of sub-regions increases. We consider the domain $\Omega = (0, 100)^2$ represented in Figure 17 where the fractures and the coarse mesh are depicted. The latter is composed by 5×5 Cartesian cells. The fractures permeability is six order of magnitude greater than the rock matrix permeability, and the fractures thickness is on the order of centimetres. To validate the upscaling procedure, the computed upscaled transmissibilities are used in an incompressible two-phase flow simulation in presence of two wells. The reservoir and fractures are completely filled with oil. We assume no-flow boundary conditions for all the reservoir. Water is pumped in at a constant rate in the injection well, I, while the production well, P is operating at a constant pressure. Both wells

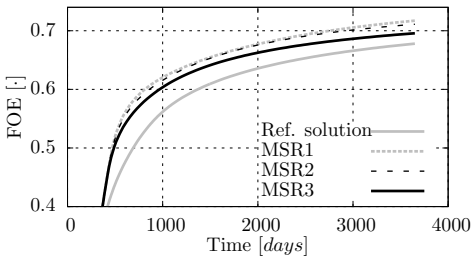

Figure 18: FOE for increasing numbers of sub-regions compared with a reference solution. MSR1 stands for multiple sub-region with one porous matrix sub-region. MSR2 and MSR3 are defined analogously.

are completed in fracture. The reference solution is computed by using EDFM with the additional matrix-matrix transmissibility obtaining a dual porosity-dual permeability model on a finer mesh comprising 100×100 Cartesian cells. The reference two-phase flow has been simulated using [22]. To compute the upscaled solutions, the upscaled transmissibilities are employed in the general purpose reservoir simulator (GPRS) [9]. In Figure 18, we show the field oil efficiency (FOE), defined as

$$FOE_t = \frac{OIP_{t=0} - OIP_t}{OIP_{t=0}},$$

where OIP is the so-called oil-in-place, which refers to the total oil content of a reservoir. It is evident that increasing the number of sub-regions the upscaled solution approaches the reference solution. However, the convergence rate is not constant through all time steps.

5.4 Comparison with a standard industrial simulator

In this section, we compare the result obtained with the upscaling procedure and [33], considered as a standard simulation tool in oil industry, against a reference solution. The upscaling considers only one porous matrix sub-region since, as shown in the previous part, increasing the number of sub-regions, we expect a better representation of the coarse solution. The test domain is represented in Figure 19, where the fractures and the 20×20 coarse mesh are represented. The permeability ratio as well as the fractures thickness are the same as in the previous test case. The reference solution is computed as in the previous test case. [33] uses the geometry-based Oda's method [30], to compute a representative permeability tensors initially defined in the discrete fracture network. The Kazemi formula is used to compute the transmissibility between the fractures and the rock matrix [25]. As in the previous test case we consider an incompressible two-phase flow simulation where the reservoir and the fractures are filled with oil. Water is pumped in at a constant rate in the injection well, I, whereas the production well, P is operating at a constant pressure. Both wells

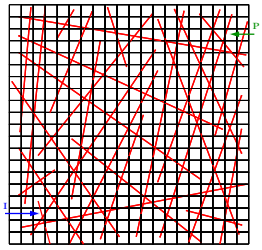

Figure 19: Coarse mesh and fractures for the comparison between the upscaling procedure and [33].

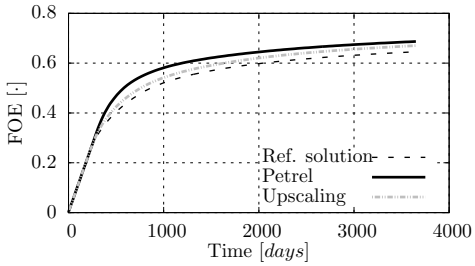

Figure 20: Comparison of the field oil efficiency for the Upscaling, [33] and the reference solution.

are completed in fracture. For the computation of the saturation field, we use [12] for both the upscaling and [33] solutions and [22] for the reference solution. The comparison of the FOE for the three approaches is reported in Figure 20. In the beginning of the simulation the three solutions are comparable. However, after 400 days the solution computed using upscaling approximates better the reference solution than the one computed using [33]. In Figure 21, the oil saturation profile for the three methods is provided at the same time level. Even though the upscaled solution shows a delay compared to the reference solution, it is able to better recover the oil dynamic than [33].

6 Conclusions

In this work, we presented a novel efficient upscaling procedure to derive a coarse scale model for a natural fractured reservoir. The main objective of the proposed scheme is to compute the upscaled parameters in an efficient and effective way. These parameters are computed via local problems which involve single or pair of coarse cells. To achieve a good efficiency, we have considered, in the solution of these problems, a non-conforming coupling between the bulk grid and the fracture grids by means of the embedded discrete fracture model method. As the numerical tests show, the upscaled problem turns out to be a reasonable and correct approximation for this class of problems. Moreover, the methodology introduced yields better results than the standard software used in

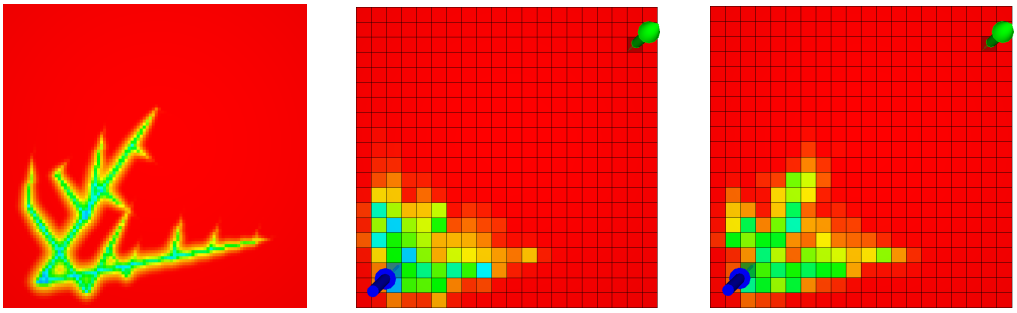

Figure 21: Representation of the oil saturation in the porous matrix at the same time level. Reference solution (left), solution obtained using [33] (center), and solution obtained using the upscaling (right).

oil industry to solve similar problems. In a forthcoming work, we will introduce a parallelization of the global algorithm to increase the efficiency and applicability to real problems.

7 Acknowledgments

The authors wish to thank Alberto Cominelli, Roberto Colin, Luca Formaggia, Matteo Longoni, Paola Panfili, Simona Perotto, Anna Scotti, and Luca Turconi for many fruitful discussions.

The General Purpose Research Simulator (GPRS) developed by the Reservoir Simulation Research Group (SUPRI-B) at Stanford University was used in this work.

A Discrete equivalence

In this section, we prove the discrete equivalence between problem (3) and (1), where the virtual fracture cell technique is used to discretize the fracture problem. Let us assume that the permeability tensor inside the rock matrix can be simplified to a scalar field, *i.e.* introducing λ, we have $\Lambda = \lambda I$. We assume also that the mobility be unitary. Moreover, we suppose that the grid fulfil the following super-admissible condition: given a cell K with one of its facet σ, we require that

$$\boldsymbol{n}_{K,\sigma} = \frac{\boldsymbol{x}_K - \boldsymbol{x}_\sigma}{d_{K,\sigma}}, \tag{11}$$

where $d_{K,\sigma}$ is the orthogonal distance between $\boldsymbol{x}_K$ and the facet σ (see [13, 14] for a detailed presentation of this kind of condition). Meshes fulfilling condition (11) are quite generals and include Cartesian grids, triangular meshes with the so called "strict Delaunay condition", *i.e.* the closure of the circumscribed circle

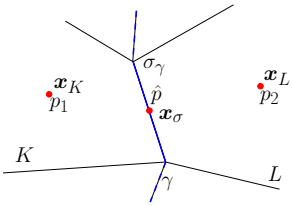

Figure 22: Schematic example of the nomenclature used to define F_σ.

to each triangle of the mesh does not contain any other triangle of the mesh, and many others. To prove the discrete equivalence, we consider one single fracture. The extension to complex networks of fractures is trivial. The notation refers to a two-dimensional problem, but an extensions to a higher dimension is possible.

The result is trivially true for cells not in contact with the fracture. Then we focus our attention on the cells close to the fracture γ, and to the fracture itself. We consider a cell of the rock matrix K, on the opposite side along the normal $\boldsymbol{n}$ of the fracture, such that one of its facets is shared with the fracture γ. We indicate this facet by σ_γ (see Figure 22 for an example of the notation). We consider a finite volume approximation of the fist equation in system (3a) in the cell K

$$
\begin{aligned}
-\int_K \nabla \cdot \lambda \nabla p_1 \mathrm{d}\boldsymbol{x} = -\sum_{\sigma \in \mathcal{E}_K} \int_\sigma \lambda \nabla p_1 \cdot \boldsymbol{n}_\sigma \mathrm{d}\boldsymbol{\sigma} = \\
= -\sum_{\sigma \in \mathcal{E}_K \setminus \sigma_\gamma} \int_\sigma \lambda \nabla p_1 \cdot \boldsymbol{n}_\sigma \mathrm{d}\boldsymbol{\sigma} - 2 \int_{\sigma_\gamma} \lambda_\gamma \left(\hat{p} - p_1\right) \mathrm{d}\boldsymbol{\sigma},
\end{aligned} \tag{12}
$$

where the first equation of system (3b) is used in the last step. We consider now another cell of the rock matrix, L, on the same side along the normal $\boldsymbol{n}$ of the fracture. We require that the cell L have σ_γ as one of its facets, so L is "on the other side of the fracture" of K. As in the case of K, we consider the integral form of the first equation in system (3a) for the cell K:

$$
\begin{aligned}
-\int_L \nabla \cdot \lambda \nabla p_2 \mathrm{d}\boldsymbol{x} = -\sum_{\sigma \in \mathcal{E}_L} \int_\sigma \lambda \nabla p_2 \cdot \boldsymbol{n}_\sigma \mathrm{d}\boldsymbol{\sigma} = \\
= -\sum_{\sigma \in \mathcal{E}_L \setminus \sigma_\gamma} \int_\sigma \lambda \nabla p_2 \cdot \boldsymbol{n}_\sigma \mathrm{d}\boldsymbol{\sigma} - 2 \int_{\sigma_\gamma} \lambda_\gamma \left(\hat{p} - p_2\right) \mathrm{d}\boldsymbol{\sigma},
\end{aligned} \tag{13}
$$

where the second equation of system (3b) is used in the last step, and we used also the fact that $\boldsymbol{n}_{\sigma_\gamma} = -\boldsymbol{n}$. The approximation by a two-point flux approximation of the facets which do not belong to the fracture in (12) and (13) is straightforward. We consider only the discretization of the last integrals of the aforementioned equations. In this case, we have to build a proper approximation of the discrete pressure at the facet σ_γ. The choice is based on the scheme

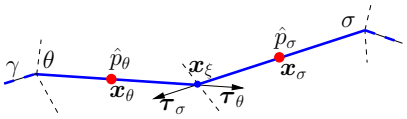

Figure 23: Schematic example of the notation used to define the flux for fracture computations.

adopted: we assume a linear variation of the pressure from the value at the centre of K to the value at the facet σ_γ. Thus we have

$$p_1|_{\sigma_\gamma} = \frac{\dfrac{\lambda}{d_{K,\sigma_\gamma}} p_1 + \dfrac{\lambda_{n,\gamma}}{d/2}\hat{p}}{\dfrac{\lambda}{d_{K,\sigma_\gamma}} + \dfrac{\lambda_{n,\gamma}}{d/2}} = \frac{\dfrac{\lambda}{d_{K,\sigma_\gamma}} p_1 + 2\lambda_\gamma \hat{p}}{\dfrac{\lambda}{d_{K,\sigma_\gamma}} + 2\lambda_\gamma}.$$

Plugging this expression in the last integral of (12), we obtain its fully discrete form

$$-2\int_{\sigma_\gamma} \lambda_\gamma \left(\hat{p} - p_1\right) \mathrm{d}\boldsymbol{\sigma} = -\left|\sigma_\gamma\right| \frac{2\lambda_\gamma \dfrac{\lambda}{d_{K,\sigma_\gamma}}}{2\lambda_\gamma + \dfrac{\lambda}{d_{K,\sigma_\gamma}}} \left(\hat{p} - p_1\right). \tag{14}$$

A similar expression can be derived for the cell L which involves p_2 as well as $\hat{p}$. Plug the expression in the last integral of (13) we obtain its fully discrete form

$$-2\int_{\sigma_\gamma} \lambda_\gamma \left(\hat{p} - p_2\right) \mathrm{d}\boldsymbol{\sigma} = -\left|\sigma_\gamma\right| \frac{2\lambda_\gamma \dfrac{\lambda}{d_{L,\sigma_\gamma}}}{2\lambda_\gamma + \dfrac{\lambda}{d_{L,\sigma_\gamma}}} \left(\hat{p} - p_2\right). \tag{15}$$

We consider now the discretization of the fracture γ by the two-point flux scheme. We consider a pair of facing facets σ and θ of the fault discretization. Unless the fracture is a straight line, we need to define two distinct unit outward normals to their common ridge ξ and, in our case, they coincide, possibly up to a sign, with the corresponding tangential vectors $\boldsymbol{\tau}_\sigma$ and $\boldsymbol{\tau}_\theta$, respectively (see Figure 23 for an example). The left-hand side of the second equation of system (3a) is discretized in the following way:

$$-\int_\xi \hat{\lambda} \nabla_{\boldsymbol{\tau}} \cdot \hat{p} \boldsymbol{n}_\xi \mathrm{d}\boldsymbol{\xi} = T_{\sigma\theta}\left(\hat{p}_\sigma - \hat{p}_\theta\right).$$

Since $\boldsymbol{x}_\sigma - \boldsymbol{x}_\xi = \boldsymbol{\tau}_\sigma \left|\sigma\right|/2$ and $\boldsymbol{x}_\theta - \boldsymbol{x}_\xi = \boldsymbol{\tau}_\theta \left|\theta\right|/2$, the half and full transmissibilities are computed by

$$T_\sigma = 2\frac{\hat{\lambda}}{|\sigma|} \quad \text{and} \quad T_\theta = 2\frac{\hat{\lambda}}{|\theta|} \quad \text{and} \quad T_{\sigma\theta} = \frac{2\hat{\lambda}}{|\sigma| + |\theta|}.$$

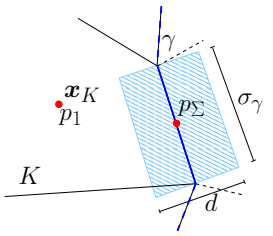

Figure 24: Schematic example of the notation used to define the flux for fracture-matrix computations in presence of virtual fracture cells.

Consider a single facet σ of the fracture discretization: the right hand-side of the second equation of system (3a) can be easily computed noticing that

$$\int_\sigma \hat{q}\mathrm{d}\boldsymbol{\sigma} + 4\int_\sigma \lambda_\gamma \left(\{\!\{p\}\!\}_\gamma - \hat{p}\right)\mathrm{d}\boldsymbol{\sigma} = \int_\sigma \hat{q}\mathrm{d}\boldsymbol{\sigma} + \\ +2\int_\sigma \lambda_\gamma \left(p_1 - \hat{p}\right)\mathrm{d}\boldsymbol{\sigma} + 2\int_\sigma \lambda_\sigma \left(p_2 - \hat{p}\right)\mathrm{d}\boldsymbol{\sigma}.$$

From equations (14) and (15) we have the balance of flux across each fracture facet. We consider now the second approach, discretizing equation (1) using the virtual fracture cells. The fracture facets are extruded by $\pm d/2$ along their normal direction, and included in the computation of the scheme. The geometrical quantities are computed locally and no modifications are perform on rock matrix cells which face the fracture, *i.e.* no volume reduction (see Figure 24 for an example of construction of virtual fracture cells). Considering a cell K facing the fracture with facet σ_γ and the associated virtual fracture cell Σ, the discretization through a two-point flux approximation gives

$$-\int_K \nabla\cdot\lambda\nabla p_1 \mathrm{d}\boldsymbol{x} = -\sum_{\sigma\in\mathcal{E}_K\setminus\sigma_\gamma}\int_\sigma \lambda\nabla p_1\cdot\boldsymbol{n}\mathrm{d}\boldsymbol{\sigma} + \\ -\int_{\sigma_\gamma}\lambda\nabla p_1\cdot\boldsymbol{n}_{\sigma_\gamma}\mathrm{d}\boldsymbol{\sigma}.$$

The summation in the previous expression is a standard application of the scheme. The last term can be discretized using the two-point approximation but with the geometrical properties computed locally. Using the super-admissible condition, the transmissibilities are

$$T_K = \lambda\frac{|\sigma_\gamma|}{d_{K,\sigma_\gamma}} \quad \text{and} \quad T_\Sigma = 2\lambda_\gamma\left|\sigma_\gamma\right|.$$

the transmissibility between the cells K and Σ gives the same expression as (14). We consider now the approximation of (1) given two virtual fracture cells Σ and Θ with common facets ξ_Σ and ξ_Θ. These cells are based on two fracture facets

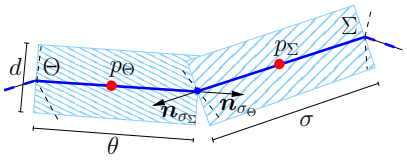

Figure 25: Schematic example of the notation used to define the flux for fracture computations in presence of virtual fracture cells.

σ and θ, respectively. We conventionally denote their common interface by σ, even though it is not well geometrically uniquely defined. In practice, σ is the representation of two distinct facets, σ_Θ and σ_Σ, with their own normals, $\boldsymbol{n}_{\sigma_\Theta}$ and $\boldsymbol{n}_{\sigma_\Sigma}$, respectively (see Figure 25 for an example). The flux between them is given by

$$-\int_\sigma \lambda \nabla p \cdot \boldsymbol{n}_\sigma \mathrm{d}\boldsymbol{\sigma} = T_{\Sigma\Theta}\left(p_\Sigma - p_\Theta\right),$$

while the transmissibilities are computed locally as

$$T_\Sigma = 2\frac{\hat{\lambda}}{|\sigma|} \quad \text{and} \quad T_\Theta = 2\frac{\hat{\lambda}}{|\theta|} \quad \text{and} \quad T_{\Sigma\Theta} = \frac{2\hat{\lambda}}{|\sigma| + |\theta|}.$$

If, in this last method, the unknowns in the fracture are indicated by $\hat{p}$, we obtain a discrete equivalence between the two approaches since the transmissibilities between correspondent unknowns are equal.

B Pseudo-steady state problem

In this section, we present a detailed and rigorous derivation of the well posedness of system (9) and the equivalence between the solution at the pseudo-steady state and the solution of a suitable stationary problem. Let Ω be an open bounded and regular subset of $\mathbb{R}^N$ with $N = 2$ or $N = 3$. We focus our attention on the following problem

$$\begin{cases} \phi c \dfrac{\partial p}{\partial t} - \nabla \cdot \dfrac{\Lambda}{\mu} \nabla p = q & \text{in } \Omega \text{ for } t > 0 \\ -\dfrac{\Lambda}{\mu} \nabla p \cdot \boldsymbol{n} = 0 & \text{on } \partial\Omega \text{ for } t > 0 \\ p = 0 & \text{in } \Omega \text{ for } t = 0, \end{cases} \tag{16}$$

where $\nu \in L^\infty(\Omega)$ and is bounded away from zero, $q \in L^2(\Omega)$ and $\boldsymbol{n}$ is the unit outward normal to $\partial\Omega$. The source function q is defined by

$$q(x) = \begin{cases} q_f & \text{if } x \in \Omega_f \\ 0 & \text{otherwise} \end{cases},$$

with $q_f \in \mathbb{R}^+$. We indicate with A the self-adjoint differential operator $Ap = -\nabla \cdot \frac{\Lambda}{\mu}\nabla p$ with domain $DA = \{u \in H^1(\Omega) : \frac{\Lambda}{\mu}\nabla p \cdot \boldsymbol{n} = 0 \text{ on } \partial\Omega\}$. Thus the differential problem (16) may be written as

$$\phi c \frac{\partial p}{\partial t} + Ap = q \quad \text{in } \Omega \text{ for } t > 0. \tag{17}$$

Thanks to Gauss's theorem and the boundary conditions, the following property holds.

Property 1. *For all $u \in DA$*

$$\int_\Omega Au = 0.$$

Moreover, we can also prove

Property 2. *The stationary problem*

$$A\overline{p} = q - C \quad \textit{with} \quad C = \frac{\phi}{\int_\Omega \phi} \int_\Omega q \tag{18}$$

admits a unique solution in the class of equivalence $H^1(\Omega)/\mathbb{R}$.

Proof. It is a straightforward application of the Fredholm alternative and the property of A. □

Property 3. *The eigenfunction problem*

$$Aw_i = \lambda_i w_i \tag{19}$$

admits a countable number of eigen-pairs (w_i, λ_i), $i = 0, 1, 2, \cdots$, with $\lambda_i \in \mathbb{R}$ and $0 = \lambda_0 < \lambda_1 < \lambda_2 < \cdots$. The eigenfunctions $w_i \in DA$ are assumed to be scaled such that $||w_i||_{L^2(\Omega)} = 1$, and $w_0 = \sqrt{|\Omega|^{-1}}$. The $\{w_i\}_{i\geq 0}$ form a Hilbert basis of DA.

Proof. It is a well known result, see [34]. □

We also observe that, from (19), the properties of the eigenvalues, Proposition 1, and the definition of w_0, it holds that

$$\int_\Omega w_i = 0, \quad i = 1, 2, \ldots, \quad \int_\Omega w_0 = \sqrt{|\Omega|}. \tag{20}$$

Let $\overline{p} \in DA$ be a particular solution of (18), for instance we may fix $\int_\Omega \overline{p} = 0$. Thanks to Property 3, we have the following result.

Property 4. *The solution of* (17) *is given by*

$$p = \overline{p} + Ct + \sum_{i=0}^{\infty} c_i \lambda_i e^{-\lambda_i t} w_i, \tag{21}$$

for certain $c_i \in \mathbb{R}$.

Proof. It is sufficient to substitute the expression (21) in (17). The c_i are determined through the initial condition, i.e., by enforcing that $p = 0$ at $t = 0$. This yields $\overline{p} + \sum_{i=0}^{\infty} c_i\, w_i = 0$, that is the c_i, for $i \geq 0$ are the unique coefficients of $-\overline{p} \in DA$ with respect to the basis $\{w_i\}_{i\geq 0}$. The convergence of the series in $H^1(\Omega)$ is thus guaranteed by the fact that $w_i \in H^1(\Omega)$. Moreover, it is a well known result that the solution is unique for $t \in [0, T)$ for any $T > 0$, see for instance [7]. □

Note that the representation of p given above is *not unique*. We have arbitrarily chosen $\overline{p}$ with zero mean. Yet, the solution is unique for $t \in [0, T)$. Of course a different choice of $\overline{p}$ will entail different values of c_i. In particular, due the choice made for $\overline{p}$, and using (20), from the relation $\overline{p} + \sum_{i=0}^{\infty} c_i\, w_i = 0$ we obtain that $c_0 = 0$ in (21).

We consider the limiting solution of (17) for $t \to \infty$. The solution exhibits a blow-up, yet we may state the following result.

Property 5. *For* $p = p(x, t)$ *solution of* (21) *we have*

$$\lim_{t\to\infty} \frac{\partial p}{\partial t} = C = \frac{\phi}{\int_\Omega \phi} \int_\Omega q.$$

Moreover

$$\lim_{t\to\infty} \frac{\Lambda}{\mu} \nabla p = \frac{\Lambda}{\mu} \nabla \overline{p} \in L^2(\Omega).$$

Proof. For the first statement it is sufficient to take the time derivative of p in (21) and recall that the λ_i, $i \geq 1$ are positive, while $c_0 = 0$. The proof of the second statement derives from the fact that A is symmetric and self-adjoint, the stability estimate of parabolic problems (theorem of Hille-Yosida) and the first statement. □

In the light of the previous results, we define $\overline{p}$ as the *pseudo-stationary* solution. Since it belongs to $H^1(\Omega)\backslash\mathbb{R}$ it is in fact known modulo a constant. The constant may be fixed by choosing, as done above, the zero mean representant. The quantity $-\dfrac{\Lambda}{\mu}\nabla\overline{p}$ represents the pseudo-stationary flux and is independent of the chosen representation too.

References

[1] Clarisse Alboin, Jérôme Jaffré, Jean E. Roberts, and Christophe Serres. Modeling fractures as interfaces for flow and transport in porous media. In *Fluid flow and transport in porous media: mathematical and numerical treatment (South Hadley, MA, 2001)*, volume 295 of *Contemp. Math.*, pages 13–24. Amer. Math. Soc., Providence, RI, 2002.

[2] Laila Amir, Michel Kern, Vincent Martin, and Jean E. Roberts. Décomposition de domaine et préconditionnement pour un modèle 3D en milieu poreux fracturé. In *Proceeding of JANO 8, 8th conference on Numerical Analysis and Optimization*, December 2005.

[3] Philippe Angot, Franck Boyer, and Florence Hubert. Asymptotic and numerical modelling of flows in fractured porous media. *M2AN Math. Model. Numer. Anal.*, 43(2):239–275, 2009.

[4] Jacob Bear. *Dynamics of Fluids in Porous Media*. American Elsevier, 1972.

[5] Stefano Berrone, Sandra Pieraccini, and S Scialò. A PDE-constrained optimization formulation for discrete fracture network flows. *SIAM Journal on Scientific Computing*, 35(2), 2013.

[6] Stefano Berrone, Sandra Pieraccini, and S Scialò. On simulations of discrete fracture network flows with an optimization-based extended finite element method. *SIAM Journal on Scientific Computing*, 35(2):908–935, 2013.

[7] Haim Brezis. *Functional analysis, Sobolev spaces and partial differential equations*. Springer, 2010.

[8] Franco Brezzi and Michel Fortin. *Mixed and Hybrid Finite Element Methods*, volume 15 of *Computational Mathematics*. Springer Verlag, Berlin, 1991.

[9] Hui Cao. *Development of techniques for general purpose simulators*. PhD thesis, Stanford University, 2002.

[10] Carlo D'Angelo and Anna Scotti. A mixed finite element method for Darcy flow in fractured porous media with non-matching grids. *Mathematical Modelling and Numerical Analysis*, 46(02):465–489, 2012.

[11] Louis J. Durlofsky. Upscaling of geocellular models for reservoir flow simulation: a review of recent progress. In *7th International Forum on Reservoir Simulation Bühl/Baden-Baden, Germany*, pages 23–27, 2003.

[12] Eclipse. Industry Reference Reservoir Simulator. `www.software.slb.com/products/foundation/pages/eclipse.aspx`, 2014.1. (accessed 28 May 2015).

[13] Robert Eymard, Thierry Gallouët, and Raphaèle Herbin. Finite volume methods. In P. G. Ciarlet and J. L. Lions, editors, *Solution of Equation in $\mathcal{R}^n$ (Part 3), Techniques of Scientific Computing (Part 3)*, volume 7 of *Handbook of Numerical Analysis*, pages 713–1018. Elsevier, 2000.

[14] Robert Eymard, Thierry Gallout, and Raphaèle Herbin. Discretization of heterogeneous and anisotropic diffusion problems on general nonconforming meshes sushi: a scheme using stabilization and hybrid interfaces. *IMA Journal of Numerical Analysis*, 30(4):1009–1043, 2010.

[15] Isabelle Faille, Alessio Fumagalli, Jérôme Jaffré, and Jean E. Roberts. Reduced models for flow in porous media containing faults with discretization using hybrid finite volume schemes. Submitted to: Computational Geosciences, 2015.

[16] Luca Formaggia, Alessio Fumagalli, Anna Scotti, and Paolo Ruffo. A reduced model for Darcy's problem in networks of fractures. *ESAIM: Mathematical Modelling and Numerical Analysis*, 48:1089–1116, 7 2014.

[17] Alessio Fumagalli and Anna Scotti. An efficient xfem approximation of darcy flow in arbitrarly fractured porous media. *Oil and Gas Sciences and Technologies - Revue d'IFP Energies Nouvelles*, 69(4):555–564, April 2014.

[18] Bin Gong, Mohammad Karimi-Fard, and Louis J. Durlofsky. An upscaling procedure for constructing generalized dual-porosity/dual-permeability models from discrete fracture characterizations. In *SPE Annual Technical Conference and Exhibition, San Antonio, Texas*. Society of Petroleum Engineers, 2006.

[19] Bin Gong, Guan Qin, Craig Douglas, and Shiyi Yuan. Detailed Modeling of the Complex Fracture Network of Shale Gas Reservoirs. *SPE Reservoir Evaluation & Engineering*, 2011.

[20] Yan Gong, Bo Li, and Zhilin Li. Immersed-interface finite-element methods for elliptic interface problems with nonhomogeneous jump conditions. *SIAM J. Numer. Anal.*, 46(1):472–495, 2008.

[21] M.-H. Hui, Bing Gong, Mohammad Karimi-Fard, and Luis J. Durlofsky. Development and Application of New Computational Procedures for Modeling Miscible Gas Injection in Fractured Reservoirs. In *SPE Annual Technical Conference and Exhibition, 11-14 November, Anaheim, California, U.S.A.* Society of Petroleum Engineers, 2007.

[22] Intersect. High-Resolution Reservoir Simulator. `www.software.slb.com/products/foundation/Pages/intersect.aspx`, 2014.1. (accessed 28 May 2015).

[23] Mohammad Karimi-Fard, Luis J. Durlofsky, and K. Aziz. An Efficient Discrete-Fracture Model Applicable for General-Purpose Reservoir Simulators. *SPE Journal*, 9(2):227–236, 2004.

[24] Mohammad Karimi-Fard, Bin Gong, and Luis J. Durlofsky. Generation of coarse-scale continuum flow models from detailed fracture characterizations. *Water Resources Research*, 42(10):n/a–n/a, 2006.

[25] H. Kazemi, L. S. Merrill Jr., K. L. Porterfield, and P. R. Zeman. Numerical Simulation of Water-Oil Flow in Naturally Fractured Reservoirs. *Society of Petroleum Engineers Journal*, 16(06):317–326, 1976.

[26] Liyong Li and Seong H. Lee. Efficient field-scale simulation of black oil in a naturally fractured reservoir through discrete fracture networks and homogenized media. *SPE Reservoir Evaluation & Engineering*, 11:750–758, 2008.

[27] Bradley T. Mallison, Mun-Hong Hui, and Wayne Narr. Practical Gridding Algorithms for Discrete Fracture Modeling Workflows. In *Ecmor XII*, number September 2010, 2010.

[28] Vincent Martin, Jérôme Jaffré, and Jean E. Roberts. Modeling fractures and barriers as interfaces for flow in porous media. *SIAM J. Sci. Comput.*, 26(5):1667–1691, 2005.

[29] H Mustapha. A new approach to simulating flow in discrete fracture networks with an optimized mesh. *SIAM Journal on Scientific Computing*, 29(4):1439–1459, 2007.

[30] M. Oda. Permeability tensor for discontinuous rock masses. *Geotechnique*, 35(4):483—-495, 1985.

[31] Paola Panfili and Alberto Cominelli. Simulation of miscible gas injection in a fractured carbonate reservoir using an embedded discrete fracture model. In *Abu Dhabi International Petroleum Exhibition and Conference, 10-13 November, Abu Dhabi, UAE*. Society of Petroleum Engineers, 2014.

[32] Paola Panfili, Alberto Cominelli, and Anna Scotti. Using Embedded Discrete Fracture Models (EDFMs) to Simulate Realistic Fluid Flow Problems. In *Second EAGE Workshop on Naturally Fractured Reservoirs, Muscat, Oman*, 2013.

[33] Petrel. E&P Software Platform. `www.software.slb.com/products/platform/Pages/petrel.aspx`, 2014.1. (accessed 28 May 2015).

[34] Roger Temam. *Navier-Stokes equations and nonlinear functional analysis*, volume 66. Siam, 1995.

[35] Antoon P. van Heel, Paulus M. Boerrigter, and Johan J. van Dorp. Thermal and Hydraulic Matrix-Fracture Interaction in Dual-Permeability Simulation. *SPE Reservoir Evaluation & Engineering*, 11(4):735–749, August 2008.

[36] J. E. Warren and P. J. Root. The behavior of naturally fractured reservoirs. *Society of Petroleum Engineers Journal*, 3(03):245–255, 1963.