Connectivity-consistent mapping method for 2-D discrete fracture networks
Delphine Roubinet, Jean-Raynald De Dreuzy, Philippe Davy

To cite this version:

HAL Id: insu-00604949
https://hal-insu.archives-ouvertes.fr/insu-00604949
Submitted on 30 Jun 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L’archive ouverte pluridisciplinaire HAL, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d’enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.
Connectivity-consistent mapping method for 2D discrete fracture networks

Delphine Roubinet, Jean-Raynald de Dreuzy and Philippe Davy

Geosciences Rennes, UMR CNRS 6118, Université de Rennes I, Rennes, France

Abstract

We present a new flow computation method in 2D Discrete Fracture Networks (DFN) intermediary between the classical DFN flow simulation method and the projection onto continuous grids. The method divides the simulation complexity by solving for flows successively at a local mesh scale and at the global domain scale. At the mesh scale, flows are determined by classical DFN flow simulations and approximated by an Equivalent Hydraulic Matrix (EHM) relating heads and flow rates discretized on the mesh borders. Assembling the Equivalent Hydraulic Matrices provides for a domain-scale discretization of the flow equation. The Equivalent Hydraulic Matrices transfer the connectivity and flow structure complexities from the mesh scale to the domain scale. Compared to existing geometrical mapping or equivalent tensor methods, the EHM method broadens the simulation range of flow to all types of 2D fracture networks both below and above the Representative Elementary Volume (REV). Additional computation linked to the derivation of the mesh-scale Equivalent Hydraulic Matrices increases the accuracy and reliability of the method. Compared to DFN methods, the EHM method first provides a simpler domain-scale alternative permeability model. Second, it enhances the simulation capacities to larger fracture networks where flow discretization on the DFN structure yields system sizes too large to be solved using the most advanced multigrid and multifrontal methods. We show that the EHM method continuously moves from the DFN method to the tensor representation as a function of the mesh-scale discretization. The balance between accuracy and model simplification can be optimally controlled by adjusting the domain-scale and mesh-scale discretizations.
1. Introduction

Fractured media has been classically modeled using either Discrete Fracture Network (DFN) or Stochastic Continuum (SC) approaches [Neuman, 2005]. Both approaches have their own advantages and drawbacks [Hsieh, 1998]. First, they differ by their underlying permeability structure and their capacity of being specified by existing field data [Hsieh, 1998]. The DFN approach easily accounts for extensive fracture characterization [Cvetkovic et al., 2004; Davy et al., 2006] while the SC approach copes more consistently with hydraulic data [Ando et al., 2003]. Second, the simulation of hydraulic processes requires the development of specific methods using the DFN approach whereas only standard discretization schemes are required with the SC approach. Third, because the SC approach simplifies the fracture network structure, it is generally less computationally demanding than the DFN method. Hybrid approaches have been developed to combine the advantages of the DFN and SC approaches. Most of them use a DFN approach at the onset for building equivalent heterogeneous continuous models mapping either the smallest fractures [Lee et al., 2001] or all fractures in the case of the Fracture Continuum Model (FCM) [Botros et al., 2008; Bourbiaux et al., 1998; Jackson et al., 2002; Reeves et al., 2008; Svensson, 2001]. Fracture Continuum Models aim at benefiting both from the structure complexity of DFNs and from the simulation and computational simplicities of continuous media. The objective is often to use the FCM approximation as a basis for simulating more computationally demanding transient or multiphase flows [Bourbiaux et al., 1998; Karimi-Fard et al., 2006].

The quality of the FCM models critically depends on the derivation of the block-scale permeabilities from the DFNs, i.e. on the mapping of the fracture network onto the continuum grid. The block is considered here as the elementary cell of the continuum grid. Block-scale permeabilities are obtained either from geometrical characteristics [Botros et al., 2008; Svensson, 2001] or through block-scale
numerical simulations of flow [Jackson et al., 2002]. Potential errors stem from differences between
the derived scalar or tensor permeabilities and the effective flows within the block. They arise from
the difficulty to account for complex fracture connectivity on a broad range of scales. For mapping
based on geometrical rules, errors decrease with finer discretization whereas for mapping based on
hydraulic computation of the equivalent permeability tensor, errors increase below the
Representative Elementary Volume [Long et al., 1982]. Jackson et al. [2002] corrected part of the
latter error by using a larger simulation zone, namely the “guard zone”, designed to remove dummy
additional fracture connectivity with the sides of the block. FCMs keep the general connectivity
structure above the scale of the block but remove most of the connectivity effects at lower scales.
This results in less flow localization at the block scale and in difficulties in defining an equivalent
block permeability tensor. A simple assessment criterion of the relevance of the tensor representation
is the difference between flows on opposite block faces. They are equal in the tensor representation.
Their difference is expected to increase steeply for complex networks below the REV scale as shown
in the example of Figure 1. To avoid handling complex connectivity at the block scale, existing FCM
methods are applied either at scales close to the smallest fractures modeled [Botros et al., 2008;
Reeves et al., 2008] or at scales larger than the Representative Elementary Volume (REV)
[Durlofsky, 1991; Jackson et al., 2002]. The first methods, i.e. the methods applicable to scales close
to the smallest fracture modeled, represent permeability by a scalar or a diagonal tensor. They
require fine grids for fractured medium representation but can be highly accurate for not too dense
fracture networks [Botros et al., 2008]. The second methods, i.e. the methods applicable to scales
larger than the REV, represent permeability by an anisotropic full tensor defined by three 2D
parameters $K_{xx}, K_{yy}$ and $K_{xy}=K_{yx}$. They require the a priori knowledge of the REV and are hence more
suited to dense fracture networks. Their drawbacks are the strong homogenization of flow, their
applicability to a restricted scale range and the increase of the numerical error with the refinement of
discretization.
None of these methods applies between the scale of the smallest fractures modeled and the REV, a scale range that spans orders of magnitude for multiscale fracture networks (i.e. fracture networks for which the fracture-length distribution is a power law) [Bonnet et al., 2001; de Dreuzy et al., 2001b]. In fact, this scale range extends at least from the connectivity scale to the REV scale. The connectivity scale is the scale at which networks are just connected. It ranges from meters to kilometers [Berkowitz et al., 2000; Davy et al., 2009]. Because of the fracture transmissivity variability, the REV scale can be one to three orders of magnitude larger than the connectivity scale [Baghbanan and Jing, 2007; de Dreuzy et al., 2001a; 2002]. Extending at least from the scales contributing to connectivity to the REV scale, the scale range of fractures contributing to flow covers several orders of magnitude from the meter to the kilometer scale. For this scale range, the only available flow simulation method is the DFN method. The DFN flow simulation method, however, is limited in terms of fracture number and domain size. The limiting step arises when solving the linear system issued from the flow discretization on the network structure. With traditional system-solving methods like the conjugate gradient, limitations stemmed from computation time. However, the new numerical methods like the multifrontal or algebraic multigrid method, as implemented in UMFPACK [Davis, 2004] and HYPRE [Falgout et al., 2005], are orders of magnitude faster but require additional memory [de Dreuzy and Erhel, 2002]. Their sole limitation is the computer memory. As a rule of thumb, they can solve at most a linear system of rank one million in a couple of minutes on a personal workstation (Pentium Xeon, 3 GHz, 8 Go). Consequently, improving simulation capacities is not about speeding up the method but about enabling simulations otherwise impossible because of memory requirements. We will thus look in this paper at the numerical memory complexity rather than at the numerical time complexity. Our longer-term strategy is to use parallel computing for performing Monte-Carlo simulations while sequential individual simulations remain sequential [Erhel et al., 2009]. This ensures scalability and a minimum of parallel computing implementation.
We propose a new FCM method for the scale range where no existing FCM method is applicable. Like with the previously-cited FCM methods, the objective is to simplify the domain-scale numerical scheme and computations while keeping the complexity of the DFN structure. The new method divides the simulation complexity by solving for flows successively at the local block scale and at the global domain scale. At the block scale, flows are determined by classical DFN flow simulations and approximated by an Equivalent Hydraulic Matrix (EHM) relating heads and flow rates discretized on the mesh borders. Assembling the Equivalent Hydraulic Matrices allows for a domain-scale discretization of the flow equation. The Equivalent Hydraulic Matrices transfer the connectivity and flow structure complexities from the block scale to the domain scale. The method is similar to Boundary Element Methods [Dershowitz and Fidelibus, 1999] as it relates heads and flow rates on the block borders. As the Equivalent Hydraulic Matrices are determined at the block scale by DFN simulations, we show that the method is systematically applicable regardless of the scale, fracture density and fracture-length and transmissivity distributions. The method accuracy and complexity are given by the level of discretization of the block borders and of the domain. We call this method the Equivalent Hydraulic Matrices (EHM) method as heads and flow rates on the block borders are linearly linked by a matrix representing the block-scale hydraulic properties rather than by a scalar or a tensor permeability. This article describes the EHM method (section 2), shows its results compared to existing methods (section 3) and discusses its performance (section 4).

2. The Equivalent Hydraulic Matrices method

This section defines the EHM method. Once the domain meshed into elementary blocks, the principle of the EHM method is to express the block-scale hydraulic properties by a linear relationship between discretized flow rates and heads on the block borders. This expression will replace the scalar or tensor models used in classical FCM models. With $p_k$ as the discretization points (also called poles) of the block numbered $k$, the vector of flow rates $\phi_k$ and heads $H_k$ on these
points are related by the following linear relationship:

\[ \phi_k = A_k \cdot H_k. \] (1)

The block matrix \( A_k \) contains sub-block scale connectivity information and can be considered as the block-scale constitutive relationship. It is obtained by performing block-scale flow simulations on the DFN. Once obtained, the block-scale matrices \( A_k \) are used for simulating flow rates at the system scale by imposing the continuity of heads and flow rates across the block borders. Relationship (1) differs a priori from Darcy’s law by its relating flow rates to heads and not to head gradients. This is only a surface difference since the construction method (section 2.2) and the resulting properties of matrices \( A_k \) (Appendix A) ensure a dependence of the flow rates on head gradients.

2.1. Discretization

Discretization is made up of two parts consisting in discretization of the domain into elementary blocks (classical meshes) and discretization of block borders into poles. The first discretization consists in defining the mesh of the Fracture Continuum Model. We use hereafter a regular grid even though the EHM method can cope with irregular meshes. Each mesh cell will be called a block. The block contains a subset of the fracture network, i.e. a sub-network, the intersections of which with the block limits are denoted \( m_k \). \( m_k(i) \) is the \( i \)th intersection of block \( k \). The second discretization consists in splitting up the block borders into segments of constant length \( d_{\text{block}} \), the discretization of each border starting at the border corner. Each segment contains either zero, one or more than one fracture border intersection \( m_k(i) \). We define poles \( p_k \) as the centers of those segments containing at least one intersection (Figure 2). Segments containing no intersection with the subnetwork are disregarded. The fundamental principle of the EHM method is that all intersections contained in the same segment are set to the same hydraulic head corresponding to the head of the pole. These additional equalities reduce the number of unknowns at the cost of the approximation that close
intersections have the same hydraulic head. The accuracy of the approximation is function of the block discretization ratio \( r_{\text{block}} \) defined as the block-border discretization scale \( d_{\text{block}} \) normalized by the block face length. The coarsest discretization corresponds to \( r_{\text{block}}=100\% \) and gives a single pole by block face. It leads to a representation close to the tensor representation (Figure 2a). It is, however, not equal to a tensor. First, opposite fluxes may not be equal. Second, some faces may not be intersected by the network and thus may not have led to a pole. Finer discretizations, obtained for decreasing \( r_{\text{block}} \) values, lead to more accurate representations converging to the DFN method when all poles correspond exactly to one intersection (Figure 2b). Like in classical numerical methods, we will show in section 3 that the numerical error of the EHM method decreases monotonously with the block-border discretization ratio \( r_{\text{block}} \), i.e. when shifting from tensor-like to DFN methods.

2.2. Construction of the block-scale Equivalent Hydraulic Matrices

Equivalent Hydraulic Matrix \( A_k \) expresses the linear relationship between flows and heads on the block border discretization. More specifically, by developing relationship (1), coefficient \( A_k(i,j) \) is the contribution of the head at the \( j^{\text{th}} \) pole to the flow at the \( i^{\text{th}} \) pole:

\[
\phi_k(i) = \sum_{j=1}^{N^k_P} A_k(i,j) \cdot H_k(j). \tag{2}
\]

where \( N^k_P \) is the pole number of block \( k \) and \( \phi_k(i) \) and \( H_k(i) \) are the flow rate and head, respectively, at \( i^{\text{th}} \) pole \( p_k(i) \). \( A_k(i,j) \) is also equal to the flow rate computed at pole \( i \) by imposing a fixed head of 1 at pole \( j \) and 0 at the other ones, i.e. a fixed head of 1 for the intersections overlapped by the segment centered on pole \( j \) and 0 for the other ones. With these boundary conditions, all coefficients of column \( j \) can be simultaneously determined by a single DFN simulation (Figure 3). The construction of the full Equivalent Hydraulic Matrix requires \( N^k_P - 1 \) simulations and not \( N^k_P \), since the sum of all elements from a column of \( A_k \) is equal to zero because of flow conservation.
We underline that this method does not require any modification of the fracture network structure or any realignment of fractures. The approximation lies exclusively in equating flows and heads at the scale of the segment of the border discretization.

### 2.3. Domain-scale flow simulation

Solving the flow equation at the domain scale consists in imposing the continuity of heads and flow rates on poles $p_k$ positioned on the block faces. External head and flow rate boundary conditions are simply implemented by imposing the head in the matrix system for the fixed head values and by adding a source term for the fixed flow rates on the corresponding poles, respectively.

We note $P$ the union of all pole points $p_k$ with the convention that poles common to two or more blocks occur only once in $P$. $P$ is made up of $N^i$ poles at the interface between two blocks ($P^i$) and of $N^f$ poles at the physical limits of the domain ($P^f$). The total number of poles at the domain scale $N$ is equal to the sum of poles of types $P^i$ and $P^f$:

$$N = N^i + N^f. \quad (3)$$

With $B(j)$ as the set of blocks sharing pole $P^i(j)$ and with $q_{b,P^i(j)}$ as the flow rate at pole $P^i(j)$ from the $b^{th}$ block of $B(j)$, flow continuity writes:

$$\sum_{b \in B(j)} q_{b,P^i(j)} = 0 \forall j \in [1, N^i]. \quad (4)$$

For the $N^d$ fixed poles at the domain limit where a Dirichlet boundary condition is applied:

$$H^d = (H^d)_0. \quad (5)$$

For the $N^n$ poles on the Neumann boundary condition, the imposed flow is simply inserted in
Equations (1), (4) and (5) lead to a linear system of \( N \) equations of the \( N \) unknown heads at the poles.

The first advantage of the EHM method compared to existing Fracture Continuum Models (FCMs) is the conservation of connectivity between blocks. In fact, faces intersected by fractures contain at least one pole whereas faces without intersecting fractures do not have any pole. This prevents dummy additional connectivity between blocks [Jackson et al., 2002; Reeves et al., 2008]. The second advantage of the EHM method is the existence of block-scale discretization parameter \( r_{\text{block}} \), which can be used to tune the balance between numerical efficiency and accuracy. The third advantage of the method is the systematic convergence with discretization and its adjustment to all kinds of 2D synthetic fracture networks as will be shown in section 3. The main drawbacks of the EHM method are the necessity to perform block-scale DFN flow simulations and the specificity of the domain-scale flow simulation that precludes the use of standard softwares like MODFLOW.

### 3. Results

#### 3.1. Fracture network types

The tested networks have been chosen so that they cover a wide range of networks both above and below the REV scale, with broad and narrow length and transmissivity distributions (Table 1). Extreme cases of low and high variability are tested in order to assess the method in highly-differentiated conditions. Network types include both lattice structures (Table 2.I) and stochastic complex fracture networks (Table 2.II-4). Stochastic fracture networks are characterized by their density, orientation, length and transmissivity distributions. The domain size given by the ratio of the domain length to the minimal fracture length is denoted by \( L \) and set to 100. It means that the fracture length distribution covers two orders of magnitude. Density is fixed by the dimensionless percolation parameter \( p \), equal to the sum of the square of the fracture lengths normalized by the
domain area. $p$ is a direct measure of connectivity as it is very close to 5.6 at the percolation threshold, whatever the other fracture network characteristics [Bour and Davy, 1997]. Three density values are used for stochastic complex fracture networks and are respectively close to threshold ($p=6$) and at around two and three times the density at threshold ($p=10$ and $p=20$). For lattice structures, $p$ is close to the number of fractures within the domain and has been chosen equal to 12 and 192 for testing methods on sparse and dense lattices, respectively. Orientations are set to 0° and 30° relative to the main flow directions for the lattice structures and are uniformly distributed for the complex stochastic fracture networks. For the complex stochastic fracture networks, fracture lengths are power-law distributed [Bonnet et al., 2001] according to the following distribution function:

$$p(l) \sim l^{-a}$$

where $l$ is the fracture length, $a$ is the characteristic power-law length exponent and $p(l)$ the fracture number of length $l$. Natural values of $a$ derived from outcrops range in the interval [2.0,3.5]. Fracture transmissivity values have been chosen to be either the same for all fractures or broadly distributed according to a lognormal distribution of logarithmic standard deviation equal to 3 [Tsang et al., 1996]. Flow boundary conditions are classical gradient-like boundary conditions with fixed head on two opposite domain faces and a constant head gradient on the orthogonal faces (Figure 1a). The bottom line of Table 2 illustrates the flow distribution computed with a broad transmissivity distribution and shows the strong channeling induced by the transmissivity distribution.

### 3.2. Comparison criteria

For comparing the performance of the EHM method with other existing methods, we use an accuracy criterion and a numerical memory complexity criterion. Accuracy is defined as the mean difference between the inlet and outlet flows and their reference counterparts. The reference is obtained from
the direct simulation on the domain-scale discrete fracture network. By denoting \( \Phi_{m}^{f_i} \) and \( \Phi_{ref}^{f_i} \) the flow rates obtained respectively by the method “m” and the reference method on face \( f_i \), the comparison criterion writes:

\[
flow_{\text{error}}_{m} = \frac{1}{2} \left( \frac{\Phi_{m}^{f_i} - \Phi_{ref}^{f_i}}{\Phi_{ref}^{f_i}} + \frac{\Phi_{m}^{f_r} - \Phi_{ref}^{f_r}}{\Phi_{ref}^{f_r}} \right) \times 100 \tag{7}
\]

where \( f_i \) and \( f_r \) stand for the left and right vertical domain faces.

The memory complexity criterion is taken as the number of non-zero elements \( nnz \) of matrix \( B \) in the linear system \( Bx=b \) issued from the discretization of the flow equation at the domain scale. Even if the number of non-zero elements is not the ideal criterion, it is still better than the system size in this case where the limitation lies rather in memory requirements than in computation time. All results represent averages over 10 simulations. We have checked that for the most complex cases \( D0 \) and \( D1 \), 10 and 100 simulations give very close results. Accuracy and numerical memory complexity results are computed for several discretizations characterized by the number of blocks (domain-scale discretization) and by \( r_{block} \) (block-scale discretization).

### 3.3. Results with existing mapping and tensor methods

To assess the Equivalent Hydraulic Matrices method, we compare it with other existing methods: first with what we call the ANIS_GEO method representing permeability by a diagonal tensor derived from fracture geometrical mapping onto the blocks and used within a finite volume method [Botros et al., 2008] and second with what we call the TENSOR_SIM method representing permeability by a full tensor obtained from block-scale DFN flow simulations and used within a mixed hybrid finite element framework (Appendix B). For these two methods, the matrix permeability is fixed to \( 10^{-12} \) m/s. We use these two methods only when they are strictly applicable.
From [Botros et al., 2008], the ANIS_GEO method is applicable only if the ratio of the block length to the minimal fracture length is lower than 2.5. For the stochastic complex networks (Table 1 B0-D1), the ratio of the domain size to the minimal fracture length is \( L=100 \), requiring for the ANIS_GEO method a domain-scale discretization of at least \( 40\times40 \) blocks. As the TENSOR_SIM method relies on the full permeability tensor at the block scale, we have determined this parameter for all studied networks from the block-scale directional permeability plots (Table 3). The method is applicable only when the directional permeability is close to an ellipse [Long et al., 1982]. It is the case for networks \( A0, A2 \) and \( D0 \) (Table 3). For the other networks, transmissivity and fracture length distributions display heterogeneities that cannot be represented by a tensor at the scale of the block.

Table 4 shows the flow error as measured by (7) using the ANIS_GEO, TENSOR_SIM and EHM methods for several domain discretizations. With the ANIS_GEO method, the flow error decreases systematically from a \( 50\times50 \) to a \( 200\times200 \) domain discretization. ANIS_GEO is particularly accurate for sparse flow structures (networks with a small fracture density or with a broad transmissivity distribution). In fact, the simple summation of the fracture contributions induced by the mapping increases sub-block-scale connectivity and hence increases flow errors. Results also show that ANIS_GEO is not applicable to networks with connectivity driven by small fractures (\( 3<a<3.5 \)), yielding errors systematically larger than 41%. To be applied systematically, the geometrical projection method ANIS_GEO requires high levels of discretization involving large linear systems (Table 5). Such discretization levels can be achieved in 2D but likely not in 3D.

The TENSOR_SIM method is accurate for regular and dense structures with an error lower than 1% for network \( A0 \) (Table 4). As opposed to the ANIS_GEO method, the error decreases when the block scale increases since the block becomes closer and eventually larger than the REV [Li et al., 2009]. The main drawback of this method is its highly limited range of application. Most of the tested
networks of Table 1 did not fulfill its conditions of application.

3.4. Assessment of the EHM method

We have tested two levels of block-scale discretization of the EHM method: \( r_{\text{block}} = 10\% \) (called the most accurate method) and \( r_{\text{block}} = 25\% \) (called the least accurate method). The EHM method gives much smaller errors than those given by the geometrical and tensor methods ANIS_GEO and TENSOR_SIM (Table 4) except for \( A0 \) (dense lattice structure with uniform fracture transmissivity) and \( D0 \) (dense fracture network with uniform fracture transmissivity) with a domain discretized by 10×10 blocks and \( r_{\text{block}} = 25\% \). For these two cases, the tensor method gives smaller errors than the least accurate EHM method. In fact, the tensor method is very accurate because the REV is smaller than the block. The large errors of the least accurate EHM method are linked to the large number of fracture intersection points with the block border set to the same head, i.e. the head of the corresponding pole. The merged points are quantified by the border merging percentage \( p_{\text{border}} \) equal to the difference in percentage between the intersection point and pole numbers. \( p_{\text{border}} \) is 0% in the absence of any approximation of the block-scale discretization and increases as larger approximations are induced by the use of a smaller number of poles for the block-scale discretization. For \( A0 \) and \( D0 \) with the 10×10 domain discretization and \( r_{\text{block}} = 25\% \), \( p_{\text{border}} \) is larger than 90% and 70%, respectively (Table 6). This explains the cases where the EHM method is less accurate than the TENSOR_SIM method. For the same networks with finer domain discretizations (30×30 and 50×50 blocks), trends are reversed and the EHM method becomes more accurate than the tensor method. For lattice cases, the flow error with the EHM method is smaller than 5% for a domain discretization of 50×50 blocks.

For stochastic complex fracture networks, flow errors range from 0.11% to 180% with a majority of errors below 10% (Table 4). Errors larger than 10% affect cases \( B2 \) and \( C2 \) characterized by a coarse
discretization of 10×10 blocks and by networks with the narrowest length distribution corresponding
to $a=3.5$. The latter fracture networks have the largest number of fractures and fracture border
intersections inducing first a stronger decrease in the numerical memory complexity (Table 5), and
then larger values of point merging percentages $p_{\text{border}}$ (Table 6). In all other cases, the flow error is
smaller than 5% for a domain discretization of 50×50 blocks. With the most accurate method
corresponding to $r_{\text{block}}=10\%$ and a domain discretization of 50×50 blocks, errors range between
0.11% and 2.1%. For 9 out of the 12 test cases for which $\sigma_{\text{lnT}} = 3$ corresponds to a fracture
transmissivity distribution spanning at least 3 orders of magnitude, errors remain as low as a few
percents showing the very good performance of the EHM method for complex flow structures.

Results of Table 4 show two interesting properties of the EHM method. First, errors are not sensitive
to the fracture transmissivity distribution as shown by the comparison of the D0 and D1 cases.
Second, errors systematically decrease both with the domain discretization at constant $r_{\text{block}}$ and with
$r_{\text{block}}$ at constant domain discretization for all complex stochastic fracture networks. These properties
offer possibilities to control the error by decreasing either the domain-scale discretization in blocks
or the block-scale discretization ratio $r_{\text{block}}$. We note that all the above simulations have been
performed on the backbone. However the applicability of the EHM method is not restricted to the
backbone as shown by its good performance on infinite clusters (Table 7). Even if errors increase by
a factor of 5 from the backbone to the infinite cluster, they still remain lower than 10% with the least
accurate method ($r_{\text{block}}=10\%$) and a domain discretization of 50×50.

3.5. Flow error versus numerical memory complexity

Numerical memory complexity is taken as the number of non-zero elements in the domain-scale
linear system issued from the discretization of the flow equation ($nnz$) (Table 5). $nnz$ determines the
memory required to solve the linear system. It does not, however, take into account the computation
of the Equivalent Hydraulic Matrices at the block scale as they are not critical in terms of system size and memory requirements. With the classical ANIS_GEO and TENSOR_SIM methods, the numerical memory complexity increases quadratically with the discretization ratio. With the EHM method, the numerical memory complexity is more variable and increases more slowly. Whatever the domain discretization and the value of $r_{block}$ for complex stochastic fracture networks, EHM methods yield smaller numerical memory complexity than the DFN method except for the $B0$ case. In the latter case, the proportion of blocks crossed by a single fracture increases the numerical memory complexity without improving the accuracy.

A more advanced evaluation of the methods is proposed by comparing their error according to their numerical memory complexity (Figures 4-6). For lattice structures (Figure 4 except magenta symbols), the EHM method is orders of magnitude more accurate than the classical methods at comparable complexities except for the $A0$ case already discussed in section 3.4. Figure 4 also shows that the accuracy of the TENSOR_SIM method increases with the numerical memory complexity as discussed in section 3.3. For the dense complex stochastic fracture network of case $D0$ (Figure 4, magenta symbols), the error with the TENSOR_SIM method is smaller than the error with all other methods at very low complexity (11%) but cannot be made smaller by refining the discretization. By contrast, with the EHM method, the error is larger at small complexity but decreases to less than 1% for the highest complexities. For the stochastic complex fracture networks (Figures 5-6), errors with the EHM method decrease with the numerical memory complexity ($nnz$), with a systematic trend close to $nnz^{-1}$. Figures 4-6 show that the errors using the EHM method with $r_{block}=10\%$ and $r_{block}=25\%$ are roughly parallel in log-log plots. For the same level of error corresponding to horizontal lines in Figures 4-6, the $r_{block}=10\%$ method yields smaller numerical memory complexities than the method with $r_{block}=25\%$.

3.6. Parameter optimization
The choice of the optimal method parameters depends on the targeted accuracy, available computation time and memory and on the fracture network structure. We illustrate the methodology to determine the appropriate parameter values on the most complex fracture network presented before D1. Basically, we show in this section that the accuracy is controlled by the discretization ratio $r_{block}$ times the length of the block edge while computation time and memory requirements are controlled by the inverse of the discretization ratio ($1/r_{block}$). The approximation of the method is performed on the block-border discretization by equating the head of points belonging to the same discretization segment. The sole parameter influencing accuracy is thus the normalized segment length $d_{block}$ equal to the discretization ratio $r_{block}$ times the length of the block edge divided by the minimal fracture length. The error $error_{flow}$ defined in (7) increases monotonously with $d_{block}$ (Figure 7). Flow errors smaller than 20% are obtained for $d_{block}$ values smaller than 2. Once the segment length has been fixed by the targeted accuracy, the computation time and memory requirements are adjusted by choosing the discretization of the system in blocks controlled by the parameter $1/r_{block}$ (Figure 8). Here the computation time refers to the full time of the flow simulation including the determination of the Equivalent Hydraulic Matrices and the solution of the large system issued by the domain-scale flow discretization. Memory requirements are still taken as the number of non-zero elements in the domain-scale matrix ($nnz$). As previously said, $nnz$ decreases for coarser domain discretizations. The computation is mainly controlled by the determination of the Equivalent Hydraulic Matrices. It first sharply decreases with $1/r_{block}$ and then increases slightly. The minimum expresses an optimal distribution of computations between the domain scale and the block scale. Smaller $1/r_{block}$ values yield more numerous smaller blocks and more Equivalent Hydraulic Matrices to determine and in turn an increase of the full computation time by more than order of magnitude. Large $1/r_{block}$ values yield less numerous larger blocks which Equivalent Hydraulic Matrices take a much larger time to determine, increasing the full computation time by at least 50%.
Similar results showing the existence of the minimum have been obtained for greater number of Monte-Carlo simulations and for different fracture network structures.

4. Discussion

The principle of the Equivalent Hydraulic Matrices method is to distribute the numerical complexity among two scales, the block-scale and the domain-scale. This method introduces a reduction of the domain-scale numerical memory complexity by coarsening the block-border discretization. The approximation consists in equating heads on nearby network points. It remains local and adjusts automatically to the specific network configuration. Like the tensor and geometrical mapping methods, the EHM method increases connectivity along block interfaces but only through the introduction of shortcuts between existing paths and not through the connection of otherwise disconnected faces. Moreover, the connectivity increase is limited to the block borders and does not affect the connectivity within the block.

The EHM method is structured around the block-scale Equivalent Hydraulic Matrices, which transfer the local connectivity information from the block scale to the domain scale. The Equivalent Hydraulic Matrices are determined by the configurations of the fracture network within the blocks but do not depend on the boundary conditions. In other words, the matrices are not intrinsic medium properties like a tensor but can be used instead of the discrete fracture network in all flow contexts both above and below the Representative Elementary Volume (REV). The Equivalent Hydraulic Matrices method is still applicable below the REV due to the adjustment of the block-scale matrices to the specificity of the connectivity structures.

Because the Equivalent Hydraulic Matrices are derived from DFN computations, it is not surprising that they contain more information than the geometrical projection methods and lead to better performance at equivalent domain-scale numerical memory complexity. We express the domain-
scale numerical memory complexity by the number of non-zero elements \((nnz)\) of the linear system issued from the discretization of the flow equation. \(nnz\) is two to four orders magnitude smaller with the EHM method than with geometrical projection methods. The EHM method also displays systematically decreasing flow errors with the domain discretization and block-scale discretization parameter \(r_{block}\). This offers possibilities to find the best optimal complexity for a given error requirement. As seen in section 3.3, this is not possible with the tensor method TENSOR_SIM and it requires too fine a domain discretization with the geometrical method ANIS_GEO.

The EHM method is intermediary between the full DFN flow simulation and the tensor method. Like in the classical tensor methods \([Jackson\ et\ al.,\ 2002\]\), the method relies on block-scale DFN simulations. It is also similar to classical numerical methods from several respects. First, it expresses the relationship between flows and heads on the block borders like many numerical methods such as finite element or boundary element methods. Second, it converges to the full DFN solution when the domain discretization or the block-scale discretization increases. As a two-scale method, it shares similarities with multiscale methods like multigrid methods. It is, however, a pure bottom-up approach in the sense that the block-scale information is used at domain scale but not the other way around. From this respect, it is closer to the principle of the multiscale finite element methods \([Efendiev\ and\ Hou,\ 2007\]\) than to the principle of multigrid methods \([Wesseling,\ 2004\]\). Finally, it remains opposed to homogenization methods since the Equivalent Hydraulic Matrices strongly depend on the block-scale fracture network structure and cannot be extrapolated to other blocks or other scales.

However, EHM methods have two drawbacks, the first one being the specificity of the domain-scale simulation method that precludes the use of commonly available continuous flow simulation softwares like MODFLOW. The second drawback is the additional numerical time complexity arising from the computation of the block-scale equivalent matrices. The total numerical complexity
includes the solution of the domain-scale linear system and the computation of the Equivalent Hydraulic Matrices at the block scale. The first contribution is evaluated by the number of non-zero elements in the domain-scale linear system $nnz$ used in the previous section. The second contribution is a function of the number of block-scale simulations multiplied by the complexity of the block-scale simulations. We have chosen to retain only the first contribution to the numerical complexity for the two following reasons. First, the complexity of the domain-scale linear system is a critical constraint. Very large systems corresponding to $nnz > 10^7$ require parallel computation. While this constraint is met only for very large systems in 2D, it is current for 3D fracture networks at much smaller domain scales. Second, the EHM methods will likely be interesting for transient simulations. In fact, the computation of the EHMs will be performed only once and the complexity of the transient simulations will depend only on the domain-scale linear system complexity. The choice of both the domain discretization and the block-scale discretization parameter will be dictated by the numerical optimization, the performance of simulations through block-scale and domain-scale computations restricted to manageable sizes, and last but not least by the required accuracy.

5. Conclusion

We have presented a new mapping method for solving the flow equation in 2D discrete fracture networks. The method consists in superposing a mesh onto the fracture network and finding the relationship between heads and flows on the borders of each block of the mesh. The relationship is linear and can be expressed in matrix form, hence the name the “Equivalent Hydraulic Matrices” (EHM) method. We have shown that this linear relationship is fundamentally analog to Darcy’s law as it is equivalent to relating flows to well-chosen head gradients on block borders. The matrix coefficients can be determined by block-scale numerical simulations and express equivalent block-scale permeability between block border zones. The zones are chosen independently for each block interface and correspond to the discretization of intersection points between the fracture network and
the block border. The method is parameterized both by the block-scale discretization parameter (block-scale discretization distance divided by the characteristic block scale) and the domain discretization (the domain scale divided by characteristic block scale in each direction). The flow simulation at the domain scale is performed simply by assembling the block-scale Equivalent Hydraulic Matrices through head and flow continuity conditions.

The interest of the EHM method is to keep good approximations of both the internal block and inter-block connectivities. Discretization is performed at a local scale and adjusts automatically to local fracture network configurations. We show on a broad range of 2D fracture networks with different density, fracture length and transmissivity distributions that the relative error of the method decreases systematically with the domain discretization and the block-scale discretization parameter, allowing for a possible automatic control of the method accuracy. We also show that the relative error of the EHM method remains restricted to a few percents for a coarse domain discretization (30×30 to 50×50), whatever the network geometrical structure and the fracture transmissivity distribution. The main advantage is its applicability to all kind of network structures, whereas the tensor method can only be used for blocks larger than the Representative Elementary Volume, a too restrictive condition for general DFN simulations. Geometrical methods give results of comparable accuracy for a much larger domain discretization leading to domain-scale numerical memory complexities orders of magnitude larger than the numerical memory complexity of the EHM method. The EHM method enables large-scale 2D flow simulation networks. We intend to test its performance on 3D fracture network simulations and in transient flow contexts.
Appendices

Appendix A: Property of the Equivalent Hydraulic Matrix

With the construction method described in section 2.2, $A_k$ has several properties. First, by imposing a fixed head of 1 at pole $j$ and 0 at the other ones as boundary conditions, the flow goes into the block by $p_k(j)$ and outward through the other poles $p_k(i)$ ($i \neq j$). Considering the flow going into the block as positive and the flow going outward as negative leads to:

\[
\begin{align*}
A_k(j, j) & \geq 0 \\
A_k(i, j) & \leq 0, \quad i \neq j
\end{align*}
\] (8)

Second, for a given column $j$, all elements $A_k(i, j)$ are determined simultaneously by solving the flow equation; mass conservation implies that

\[
\sum_{i=1}^{N_p^k} A_k(i, j) = 0.
\] (9)

Or similarly:

\[
A_k(i, i) = -\sum_{j=1, j \neq i}^{N_p^k} A_k(j, i).
\] (10)

Third, because the reciprocity principle is applicable in the case of Darcian flow [Barker, 1991], $A_k$ is symmetric:

\[
A_k(i, j) = A_k(j, i).
\] (11)

Fourth, we show that the linear relationship (1) between flows and heads with property (10) leads to a relationship between flows and head gradients. In fact:
\[ \phi_k(i) = \sum_{j=1}^{N_p^k} A_k(i,j) \times H_k(j) \]

\[ \phi_k(i) = \sum_{j=1, j \neq i}^{N_p^k} A_k(i,j) \times H_k(j) + A_k(i,i) \times H_k(i) \]

and using (10):

\[ \phi_k(i) = \sum_{j=1, j \neq i}^{N_p^k} A_k(i,j) \times (H_k(j) - H_k(i)) \]

\[ \phi_k(i) = \sum_{j=1}^{N_p^k} A_k(i,j) \times x_{k,ij} \times \frac{(H_k(j) - H_k(i))}{x_{k,ij}} \] \hspace{1cm} (12)

where \( x_{k,ij} \) is the distance between poles \( p_k(i) \) and \( p_k(j) \). Equation (12) shows that flow \( \phi_k(i) \) at \( p_k(i) \) is the sum of the head gradients from \( p_k(i) \) to the other poles. Equation (12) gives a simple interpretation of \( A_k(i,j) \times x_{k,ij} \). \( A_k(i,j) \times x_{k,ij} \) is the proportionality coefficient between flow \( \phi_k(i) \) and the head gradient \( (H_k(j) - H_k(i))/x_{k,ij} \) between \( p_k(i) \) and \( p_k(j) \). \( A_k(i,j) \times x_{k,ij} \) can thus be interpreted as an “equivalent transmissivity” between the \( i^{th} \) and \( j^{th} \) poles.
Appendix B: Tensor permeability and finite elements (TENSOR_SIM method)

The Equivalent Hydraulic Matrices method consists in dividing the domain into blocks and describing block-scale hydraulic properties using Equivalent Hydraulic Matrices. The discretization of the block borders by poles $p_k$ is determined by the block-scale discretization parameter $r_{block}$, which is the ratio of the block-scale distance discretization to the block length. This parameter $r_{block}$ drives the discretization of intersections $m_k$ between block borders and fractures. Coefficients of the EHMs are determined by simulations at the block scale as described in section 2.2. The EHMs are equivalent to tensors in that they impose the following discretization and construction rules: (1) $r_{block}=100\%$, i.e. each block border is discretized by at most one pole, (2) each block border is represented by one pole (even if there is no intersection point), (3) matrix coefficients are determined by applying head gradient boundary conditions in the vertical and horizontal directions [Renard et al., 2001], (4) the computed flow rates used for the determination of the coefficients are the directional flow rates, i.e. the mean of the flow rates going out of the domain through borders perpendicular to the studied direction, and (5) coefficients are corrected to obtain symmetric positive definite tensors [Long et al., 1982]. Adding these rules of determination, the Equivalent Hydraulic Matrices become tensors that describe block-scale permeability. Computed block-scale tensors are used within a classical mixed hybrid method adapted for quadrangles to simulate flow at the domain scale [Chavent and Roberts, 1991]. We denote this method the TENSOR_SIM method.
Notations

506 $K_{xx}$ permeability in the $x$-direction due to a head gradient in the $x$-direction, m/s.

507 $K_{yy}$ permeability in the $y$-direction due to a head gradient in the $y$-direction, m/s.

508 $K_{xy}$ permeability in the $x$-direction due to a head gradient in the $y$-direction, m/s.

509 $K_{yx}$ permeability in the $y$-direction due to a head gradient in the $x$-direction, m/s.

510 $p_k$ vector of poles.

511 $\Phi_k$ vector of flow rates at the poles for block $k$, m$^2$/s.

512 $H_k$ vector of heads at the poles for block $k$, m.

513 $A_k$ Equivalent Hydraulic Matrix of block $k$, m/s.

514 $m_k$ vector of intersections between the fractures and the faces of block $k$.

515 $d_{block}$ discretization distance of block borders, m.

516 $r_{block}$ discretization ratio of block borders.

517 $N_{\phi}^k$ number of poles of block $k$.

518 $x_{k,ij}$ distance between the $i^{th}$ and $j^{th}$ poles, m.

519 $P$ union of all poles.
union of poles on block interfaces.

union of poles on domain faces.

total number of poles.

number of poles of type $P^i$.

number of poles of type $P^f$.

set of blocks sharing pole $P^i(j)$

flow rate at poles $P^i(j)$ from the $b^{th}$ block, m$^2$/s.

number of poles on the Dirichlet boundary condition.

number of poles on the Neumann boundary condition.

head of poles on the Neumann boundary condition, m

fixed head on the Neumann boundary condition, m

percolation parameter.

fracture length, m.

fracture length distribution.

power law exponent.
flow rate computed by the method “m” on the face $f_i$, m$^2$/s.

flow rate computed by the reference method on the face $f_i$, m$^2$/s.

number of non-zero elements of the domain-scale linear system.

border discretization percentage
Acknowledgments

This work was supported by the ANR project MICAS. We thank Jocelyne Erhel for fruitful discussions.
Bibliography


Figure captions

Figure 1 – Fracture network at the block scale (a) and corresponding flows (b) for the gradient head boundary conditions illustrated in (a). Fracture network parameters are the system size $L$ normalized by the smallest fracture length ($L_{\text{normal}}=100$), the fracture density number twice larger as its value at percolation threshold, the power-law fracture length exponent of 2.5 and the lognormal transmissivity distribution of logarithmic standard deviation 3.0. Boundary flows integrated on the domain sides and normalized by the mean fracture transmissivity are given in (b). They display large differences between opposite sides and illustrate the non-tensor nature of the flows.

Figure 2 – Principle of the block-border discretization with two different discretization scales $d_{\text{block}}$ corresponding to the side length (a) and to half of it (b). The backbone of the sub-network contained in the block is represented by the grey segments. Intersections $m_k$ between the backbone and the block borders are the black dots. Discretization segments and poles $p_k$ are respectively the color dashed segments and crosses. In (a), the four discretization segments intersect the backbone in one or two points. The four poles corresponding to the four crosses are thus defined and the Equivalent Hydraulic Matrix (EHM) is of rank 4. In (b), only six of the eight discretization segments intersect the backbone leading to the definition of 6 poles and to an EHM of rank 6.

Figure 3 – Principle of the determination of one of the columns of the Equivalent Hydraulic Matrix $A_k$. In this example, block $k$ is made up of fives intersections between the sub-network and the block borders (black points) and four poles (blue crosses). The boundary conditions applied to poles illustrated in (a) are a fixed head of 1 for the 2nd pole and 0 for the other ones. They condition the boundary conditions applied to the intersections illustrated in (b), which are a fixed head of 1 for the
intersections represented by the 2\textsuperscript{nd} pole and 0 for the other ones. Flow rates in poles (d) are deduced from flow rates at the intersections (c). The flow rate at the \(i\)\textsuperscript{th} pole is the sum of the flow rates at the intersections represented by this pole. The elements of the second column of the matrix \(A_k\) are deduced from flow rates computed at the poles (e).

Figure 4 – flow\_error versus numerical memory complexity (nnz) for lattice structures and dense fracture networks with constant fracture transmissivity (magenta symbols). The grey area underlines a lower part of the graph where errors range between \(5 \times 10^{-6}\%\) and \(10^{-4}\%\). The dashed horizontal line pictures the 10% error value. The dashed diagonal lines are power-law functions of exponent -1 and are meant as a guide for the eye for the decrease tendency of the EHM method. Note that errors larger than \(10^3\) are not represented.

Figure 5 – flow\_error versus numerical memory complexity (nnz) for stochastic complex fracture networks at threshold with distributed fracture transmissivities. The dashed horizontal line pictures the 10% error value. The dashed diagonal lines are power-law functions of exponent -1 and are meant as a guide for the eye for the decrease tendency of the EHM method. Note that errors larger than \(10^3\) are not represented.

Figure 6 – flow\_error versus numerical memory complexity (nnz) for stochastic complex fracture networks with distributed fracture transmissivities. The dashed horizontal line pictures the 10% error value. The dashed diagonal lines are power-law functions of exponent -1 and are meant as a guide for the eye for the decrease tendency of the EHM method. Note that errors larger than \(10^3\) are not represented.

Figure 7 – flow\_error versus \(d_{\text{block}}\) the discretization ratio \(r_{\text{block}}\) times the length of the block edge for the network \(D1\) (domain size \(L=100\)).
Figure 8 – Computation time (red dashed line) and numerical memory complexity taken as the number of non-zero elements in the largest matrix (black line) as a function of block size divided by the segment discretization length $1/r_{\text{block}}$ for $D1$ with $d_{\text{block}}$ equal to 1.
Table 1 – Characteristics of the tested fracture networks. The first four networks are on-lattice structures whereas the other ones are off-lattice structures. The ratio of the domain size $L$ to the length of the smallest fracture $l_{\min}$ is set to 100. The fracture density is characterized by the percolation parameter $p$ [Bour and Davy, 1998]. Fractures either cross the whole domain ($a=1$) or are distributed according to a power-law distribution ($a>1$). Fracture orientations ($\theta$) are either specified to a set of fixed angles (first four cases) or uniformly distributed. Fracture transmissivity is constant ($\sigma_{lnT} = 0$) or lognormally distributed with a lognormal standard deviation ($\sigma_{lnT} = 3$). The last column indicates the figure numbers displaying an example of the network type in Table 2. Fracture network types are classified in family of networks: “A” is for lattice structures, “B” for networks at percolation threshold, “C” for networks with an intermediary fracture density and “D” for dense networks.
Table 2 – Illustration of the tested fracture networks with examples of backbones (I1-I4), infinite clusters (I4’) and flows (II1-II4) obtained with the gradient boundary conditions illustrated by Figure 1a and computed with a broadly distributed fracture transmissivity $\sigma_{\text{INT}} = 3$ (see Table 1). Flow magnitude is represented by grey intensity and segment width. I1, I2, I3 and I4-4’ correspond to network types A3, B1, C2 and D1 (Table 1). Red squares stand for an elementary block corresponding to a domain discretization of 10×10 blocks.
Table 3 – Polar plots of permeability for the networks of Table 1, representing the permeability versus the polar angle $\theta$. Each point represents for a given $\theta$ the permeability computed in a square of side length $L/3$ (where $L$ is the domain size), of axis rotated by $\theta$ and centered on the initial system center.
<table>
<thead>
<tr>
<th>Domain discretization</th>
<th>ANIS_GEO</th>
<th>TENSOR_SIM</th>
<th>EHM (10%)</th>
<th>EHM (25%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10×10</td>
<td>30×30</td>
<td>50×50</td>
<td>200×200</td>
</tr>
<tr>
<td></td>
<td>10×10</td>
<td>30×30</td>
<td>50×50</td>
<td>10×10</td>
</tr>
<tr>
<td></td>
<td>10×10</td>
<td>30×30</td>
<td>50×50</td>
<td>10×10</td>
</tr>
<tr>
<td></td>
<td>10×10</td>
<td>30×30</td>
<td>50×50</td>
<td>10×10</td>
</tr>
<tr>
<td></td>
<td>10×10</td>
<td>30×30</td>
<td>50×50</td>
<td>10×10</td>
</tr>
</tbody>
</table>

Table 4 – flow_error as defined by equation (7) for the backbone of the fracture network types defined in Table 1 and for the three computational methods ANIS_GEO, TENSOR_SIM and EHM at different discretization levels. Domain discretization refers to the ratio of the domain size to the block size in each direction. EHM methods are characterized in brackets by their block-scale discretization parameter \( r_{\text{block}} \) equal to the ratio expressed in % between the block-scale discretization distance \( d_{\text{block}} \) and the block scale. Empty cells mean that the conditions of application of the method are not fulfilled in the corresponding case.
<table>
<thead>
<tr>
<th>Domain discretization</th>
<th>DFN</th>
<th>ANIS_GEO</th>
<th>TENSOR_SIM</th>
<th>EHM (10%)</th>
<th>EHM (25%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>10×10</td>
<td>30×30 50×50</td>
<td>200×200</td>
<td>10×10 30×30 50×50</td>
</tr>
<tr>
<td>A0</td>
<td>26</td>
<td>0.46</td>
<td>4.4</td>
<td>12</td>
<td>200</td>
</tr>
<tr>
<td>A1</td>
<td>26</td>
<td>0.46</td>
<td>4.4</td>
<td>12</td>
<td>200</td>
</tr>
<tr>
<td>A2</td>
<td>0.16</td>
<td>0.46</td>
<td>4.4</td>
<td>12</td>
<td>200</td>
</tr>
<tr>
<td>A3</td>
<td>0.16</td>
<td>0.46</td>
<td>4.4</td>
<td>12</td>
<td>200</td>
</tr>
<tr>
<td>B0</td>
<td>0.87</td>
<td></td>
<td>12</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>B1</td>
<td>13</td>
<td></td>
<td>12</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>B2</td>
<td>33</td>
<td></td>
<td>12</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>C0</td>
<td>6.1</td>
<td></td>
<td>12</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>C1</td>
<td>53</td>
<td></td>
<td>12</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>C2</td>
<td>240</td>
<td></td>
<td>12</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>D0</td>
<td>52</td>
<td></td>
<td>12</td>
<td>200</td>
<td></td>
</tr>
<tr>
<td>D1</td>
<td>51</td>
<td></td>
<td>12</td>
<td>200</td>
<td></td>
</tr>
</tbody>
</table>

Table 5 – Numerical memory complexity expressed as the number of non-zero elements ($nnz$) of the domain-scale linear systems issued from the flow discretization for the network cases of Table 1. Parameters are identical to those of Table 4. All numbers are expressed in thousands of non-zero elements.
<table>
<thead>
<tr>
<th>Domain discretization</th>
<th>EHM (10%)</th>
<th>EHM (25%)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>10×10</td>
<td>30×30</td>
</tr>
<tr>
<td>A0</td>
<td>70</td>
<td>21</td>
</tr>
<tr>
<td>A1</td>
<td>69</td>
<td>23</td>
</tr>
<tr>
<td>A2</td>
<td>4.7</td>
<td>1.7</td>
</tr>
<tr>
<td>A3</td>
<td>4.7</td>
<td>1.7</td>
</tr>
<tr>
<td>B0</td>
<td>14</td>
<td>7</td>
</tr>
<tr>
<td>B1</td>
<td>44</td>
<td>23</td>
</tr>
<tr>
<td>B2</td>
<td>54</td>
<td>32</td>
</tr>
<tr>
<td>C0</td>
<td>25</td>
<td>12</td>
</tr>
<tr>
<td>C1</td>
<td>53</td>
<td>29</td>
</tr>
<tr>
<td>C2</td>
<td>72</td>
<td>44</td>
</tr>
<tr>
<td>D0</td>
<td>50</td>
<td>26</td>
</tr>
<tr>
<td>D1</td>
<td>50</td>
<td>26</td>
</tr>
</tbody>
</table>

Table 6 – Block-scale border merging percentage $p_{\text{border}}$ for the fracture network types of Table 1.
<table>
<thead>
<tr>
<th>Domain discretization</th>
<th>ANIS_GEO</th>
<th>TENSOR_SIM</th>
<th>EHM (10%)</th>
<th>EHM (25%)</th>
<th>Table 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>10×10</td>
<td>5.5×10^2</td>
<td>37</td>
<td>11</td>
<td>30×30</td>
<td>23</td>
</tr>
<tr>
<td>30×30</td>
<td>1.5×10^2</td>
<td>38</td>
<td>4.9</td>
<td>50×50</td>
<td>14</td>
</tr>
<tr>
<td>50×50</td>
<td>200×200</td>
<td>14</td>
<td>3.0</td>
<td>10×10</td>
<td>76</td>
</tr>
<tr>
<td>10×10</td>
<td>1.7×10^3</td>
<td>12</td>
<td>4.3</td>
<td>30×30</td>
<td>70</td>
</tr>
<tr>
<td>30×30</td>
<td>1.5×10^2</td>
<td>2.4</td>
<td>20</td>
<td>50×50</td>
<td>11</td>
</tr>
</tbody>
</table>

Table 7 – flow_error as defined by equation (7) for the infinite cluster of fracture networks D0 and D1 (see Table 1 for description). EHM methods are characterized in brackets by their block-scale discretization parameter r_{block} equal to the ratio expressed in % of the block-scale discretization distance d_{block} to the block scale.
Figure 1 – Fracture network at the block scale (a) and corresponding flows (b) for the gradient head boundary conditions illustrated in (a). Fracture network parameters are the system size $L$ normalized by the smallest fracture length ($L = 100$), the fracture density number twice larger as its value at percolation threshold, the power-law fracture length exponent of 2.5 and the lognormal transmissivity distribution of logarithmic standard deviation 3.0. Boundary flows integrated on the domain sides and normalized by the mean fracture transmissivity are given in (b). They display large differences between opposite sides and illustrate the non-tensor nature of the flows.
Figure 2 – Principle of the block-border discretization with two different discretization scales $d_{\text{block}}$ corresponding to the side length (a) and to half of it (b). The backbone of the sub-network contained in the block is represented by the grey segments. Intersections $m_k$ between the backbone and the block borders are the black dots. Discretization segments and poles $p_k$ are respectively the color dashed segments and crosses. In (a), the four discretization segments intersect the backbone in one or two points. The four poles corresponding to the four crosses are thus defined and the Equivalent Hydraulic Matrix (EHM) is of rank 4. In (b), only six of the eight discretization segments intersect the backbone leading to the definition of 6 poles and to an EHM of rank 6.
Figure 3 – Principle of the determination of one of the columns of the Equivalent Hydraulic Matrix $A_k$. In this example, block $k$ is made up of five intersections between the sub-network and the block borders (black points) and four poles (blue crosses). The boundary conditions applied to poles illustrated in (a) are a fixed head of 1 for the 2$^{nd}$ pole and 0 for the other ones. They condition the boundary conditions applied to the intersections illustrated in (b), which are a fixed head of 1 for the intersections represented by the 2$^{nd}$ pole and 0 for the other ones. Flow rates in poles (d) are deduced from flow rates at the intersections (c). The flow rate at the $i^{th}$ pole is the sum of the flow rates at the intersections represented by this pole. The elements of the second column of the matrix $A_k$ are deduced from flow rates computed at the poles (e).
Figure 4 – *flow_error* versus numerical memory complexity (*nnz*) for lattice structures and dense fracture networks with constant fracture transmissivity (magenta symbols). The grey area underlines a lower part of the graph where errors range between $5 \times 10^{-6}\%$ and $10^{-4}\%$. The dashed horizontal line pictures the 10% error value. The dashed diagonal lines are power-law functions of exponent -1 and are meant as a guide for the eye for the decrease tendency of the EHM method. Note that errors larger than $10^3$ are not represented.
Figure 5 – flow_error versus numerical memory complexity (nnz) for stochastic complex fracture networks at threshold with distributed fracture transmissivities. The dashed horizontal line pictures the 10% error value. The dashed diagonal lines are power-law functions of exponent -1 and are meant as a guide for the eye for the decrease tendency of the EHM method. Note that errors larger than $10^3$ are not represented.
Figure 6 – *flow_error* versus numerical memory complexity (*nnz*) for stochastic complex fracture networks with distributed fracture transmissivities. The dashed horizontal line pictures the 10% error value. The dashed diagonal lines are power-law functions of exponent -1 and are meant as a guide for the eye for the decrease tendency of the EHM method. Note that errors larger than $10^3$ are not represented.
Figure 7 – flow_error versus $d_{block}$ the discretization ratio $r_{block}$ times the length of the block edge for the network $D1$ (domain size $L=100$).
Figure 8 – Computation time (red dashed line) and numerical memory complexity taken as the number of non-zero elements in the largest matrix (black line) as a function of block size divided by the segment discretization length $1/r_{block}$ for D1 with $d_{block}$ equal to 1.