HAL open science

Combination of HHO and Mesh refining/Coarsening Strategies for Efficient Flow Solvers in Fractured Rocks

Géraldine Pichot, Florent Hédin, Nicolas Pignet, Alexandre Ern

To cite this version:

Géraldine Pichot, Florent Hédin, Nicolas Pignet, Alexandre Ern. Combination of HHO and Mesh refining/Coarsening Strategies for Efficient Flow Solvers in Fractured Rocks. GS21 - SIAM Conference on Mathematical and Computational Issues in the Geosciences, Jun 2021, Milan / Virtual, Italy. hal-03519924

HAL Id: hal-03519924

https://inria.hal.science/hal-03519924

Submitted on 10 Jan 2022

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.

Combination of HHO and Mesh refining/Coarsening Strategies for Efficient Flow Solvers in Fractured Rocks.

Géraldine Pichot[1], Florent Hédin[1], Nicolas Pignet[3] and Alexandre Ern[2]

[1]*Inria SERENA & ENPC*

[2] *ENPC & Inria SERENA*

[3]*EDF*

Work in collaboration with

Patrick Laug (Inria GAMMA3, Saclay)

Simon Legrand (Inria SED Paris)

Romain Le Goc, Caroline Darcel (ITASCA S.A.S)

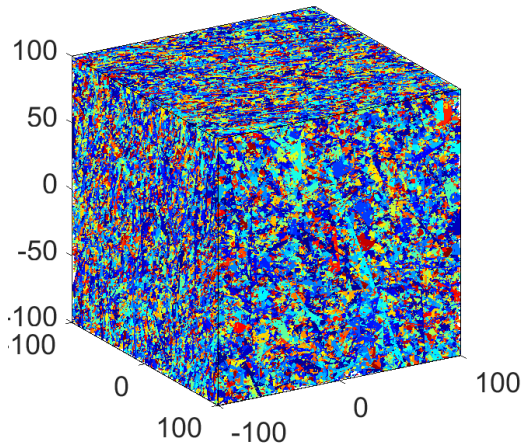
and Philippe Davy (Geosciences Rennes, CNRS)

Context

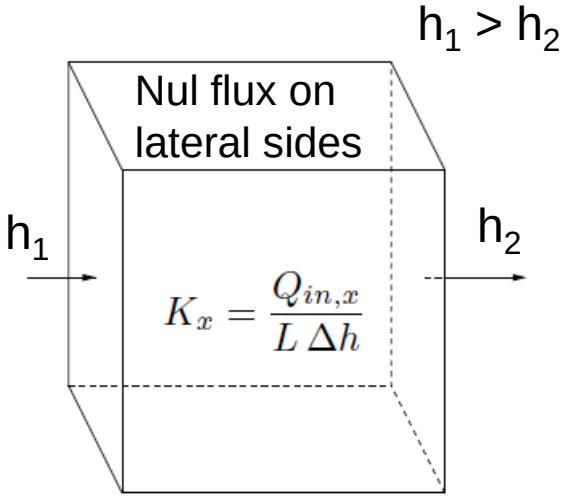
Context

- **Collaborative work with the French ANR LabCom *fractory* (ITASCA S.A.S & CNRS)**
  - ✓ Realistic DFN modeling: Unified Fracture Model (UFM)
  - ✓ Realistic physical properties (one transmissivity value per fracture)

Davy, P., R. Le Goc, and C. Darcel (2013), A model of fracture nucleation, growth and arrest, and consequences for fracture density and scaling, *Journal of Geophysical Research: Solid Earth*, *118*(4), 1393-1407



L=200 m
1,176,566 fractures



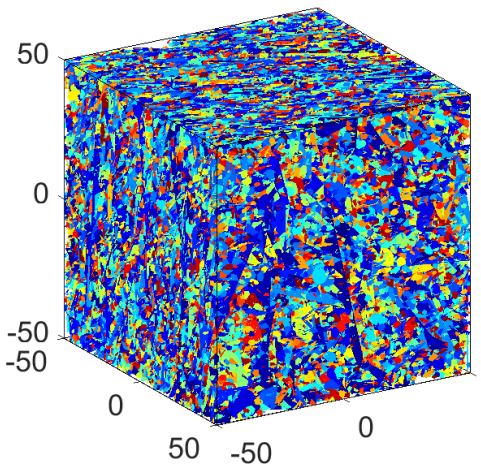
Permeameter BCs

- **Challenging objective: efficient flow simulations in networks with thousands to millions of fractures (~10^3 to ~10^6)**

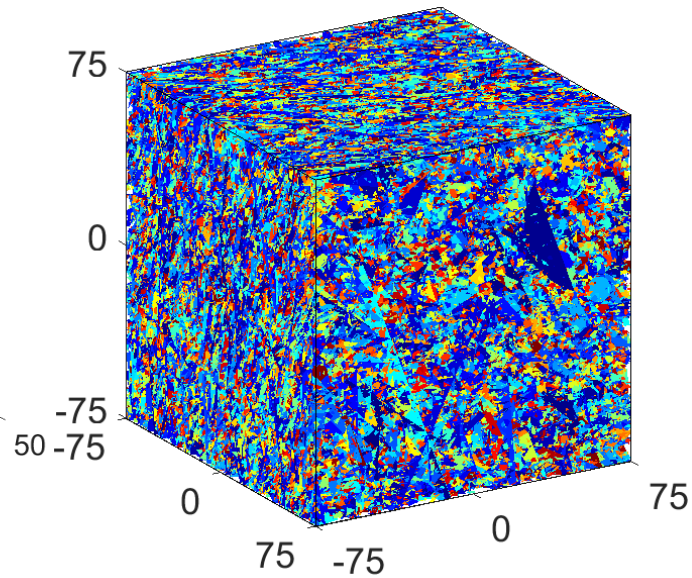
- **Computation of macroscopic quantities of interest:** estimation of the diagonal components of the equivalent permeability tensor for large DFN embedded in a cubic domain of size L (L=200m) with **permeameter** test cases.

Context: test case provided by the LabCom fractory

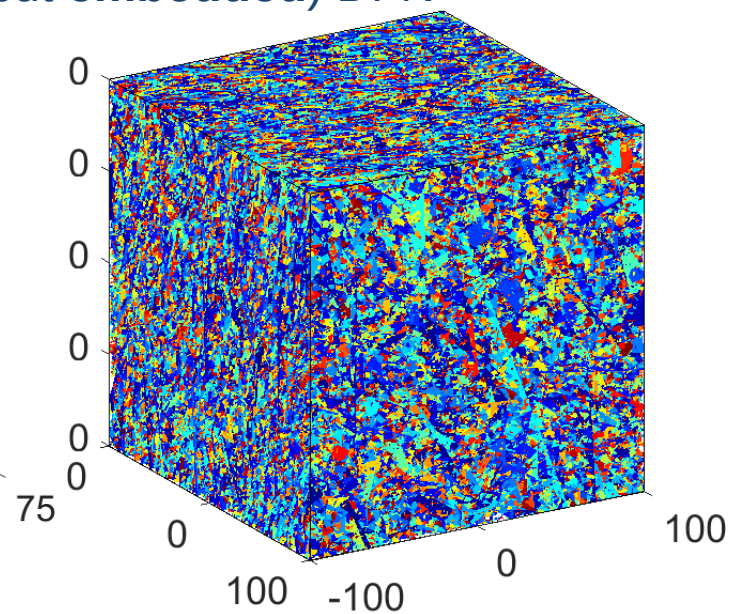
- We change the cube size L to work on different (but **embedded**) DFN



L=100 m



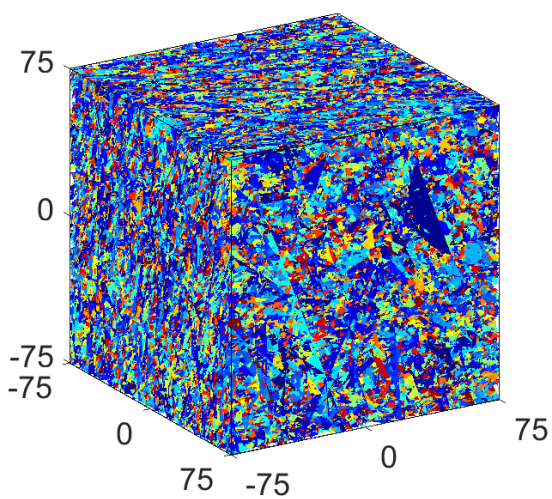
L=150 m

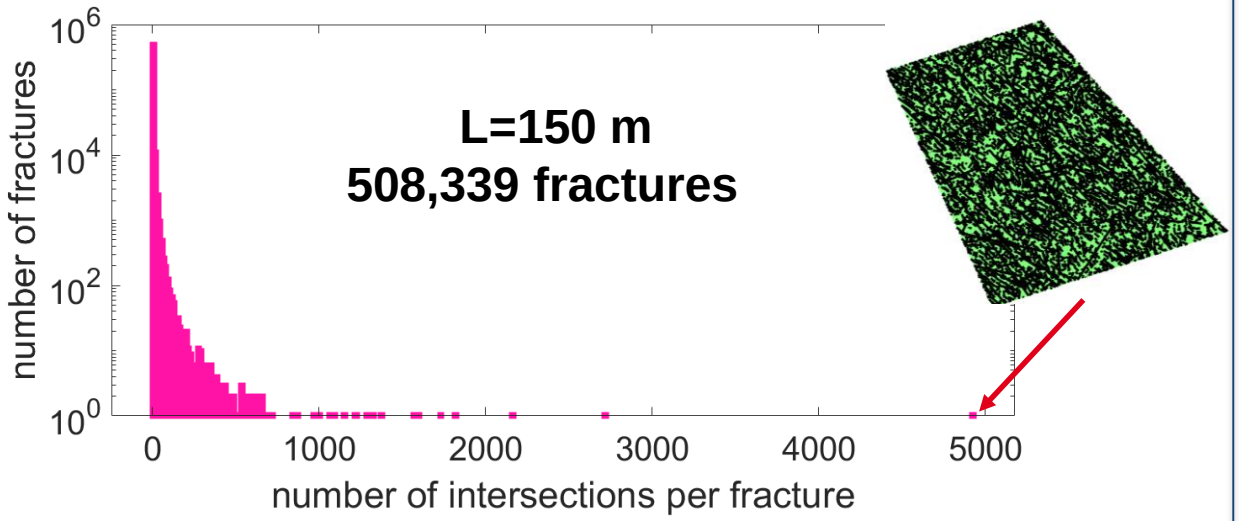


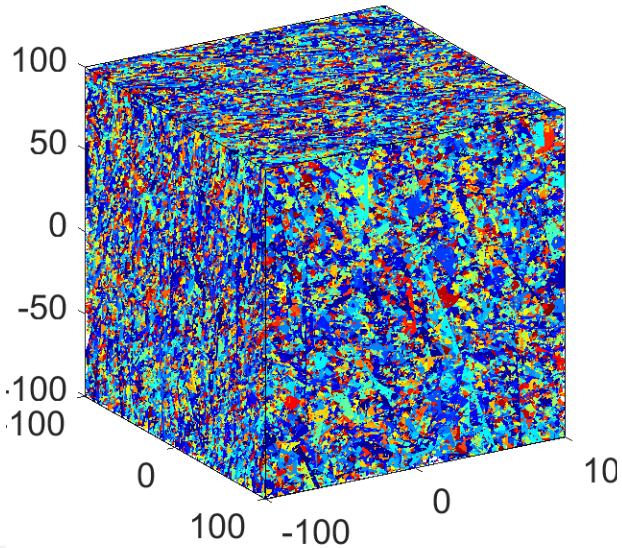
L=200 m

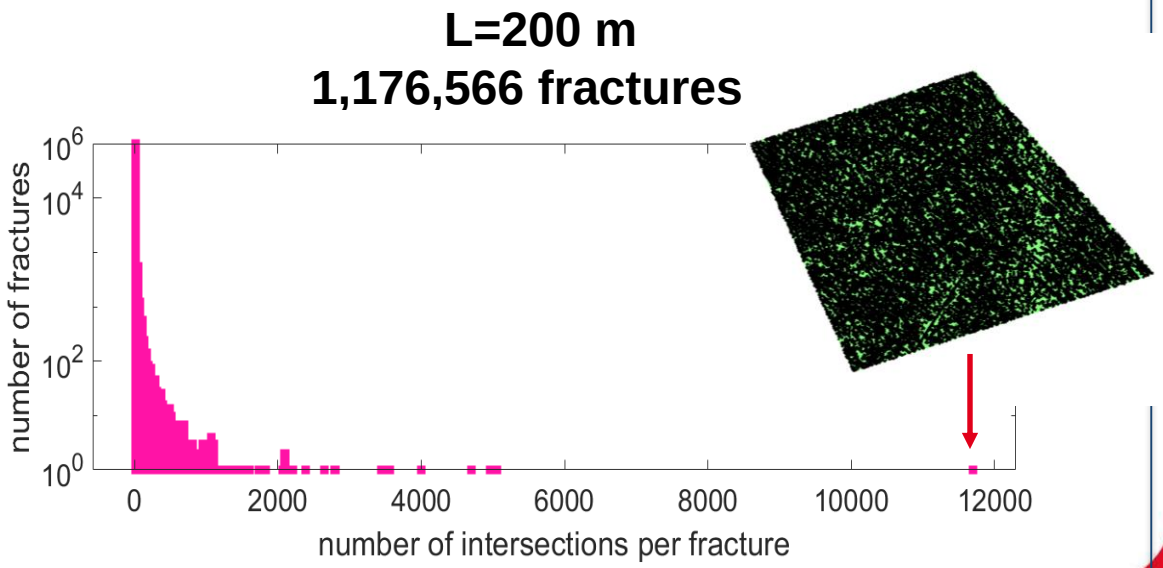
Cube size L	# fractures	# intersections	Max(#inter per fracture)	Transmissivity range [$m^2.s^{-1}$]
100	152,405	302,907	1,022	[3.5e-06; 20.33]
150	508,339	1,031,231	4,930	[3.38-06; 25.8]
200	**1,176,566**	2,410,539	11,710	[3.35-06; 25.8]

Context: test case provided by the LabCom fractory









The flow problem

J. Erhel, J.-R. de Dreuzy, and B. Poirriez. SIAM J. Sci. Comput. (2009)

J. Maryska, O. Severyn, and M. Vohralik, Computational Geosciences (2004)

Assumptions:

- The rock matrix is impervious: flow is only simulated in the fractures
- Study of steady state flow
- There is no longitudinal flux in the intersections of fractures

Flow equations within each fracture γ

$$div_\gamma \mathbf{u}(\mathbf{x}) = \mathbf{f}(\mathbf{x}), \quad \text{for } \mathbf{x} \in \gamma, \quad \textit{(Continuity equation)}$$

$$\mathbf{u}(\mathbf{x}) = -\mathbf{T}_\gamma(\mathbf{x})\nabla p(\mathbf{x}), \quad \text{for } \mathbf{x} \in \gamma, \quad \textit{(Poiseuille's law)}$$

$$p(\mathbf{x}) = p^D(\mathbf{x}), \quad \text{on } \Gamma_D \cap \Gamma_\gamma, \quad \textit{(Dirichlet boundary conditions)}$$

$$\mathbf{u}(\mathbf{x}).\nu = q^N(\mathbf{x}), \quad \text{on } \Gamma_N \cap \Gamma_\gamma, \quad \textit{(Neumann boundary conditions)}$$

$$\mathbf{u}(\mathbf{x}).\mu = 0, \quad \text{on } \Gamma_\gamma \backslash \{(\Gamma_\gamma \cap \Gamma_D) \cup (\Gamma_\gamma \cap \Gamma_N)\}, \quad \textit{(Condition of an impervious matrix)}$$

- ν (resp. μ) outward normal unit vectors
- $\mathbf{T}_\gamma(\mathbf{x})$ a given transmissivity field (unit $[\mathrm{m}^2.\mathrm{s}^{-1}]$), $\mathbf{f}(\mathbf{x}) \in L^2(\Omega_f)$ sources/sinks.

Continuity conditions at each intersection

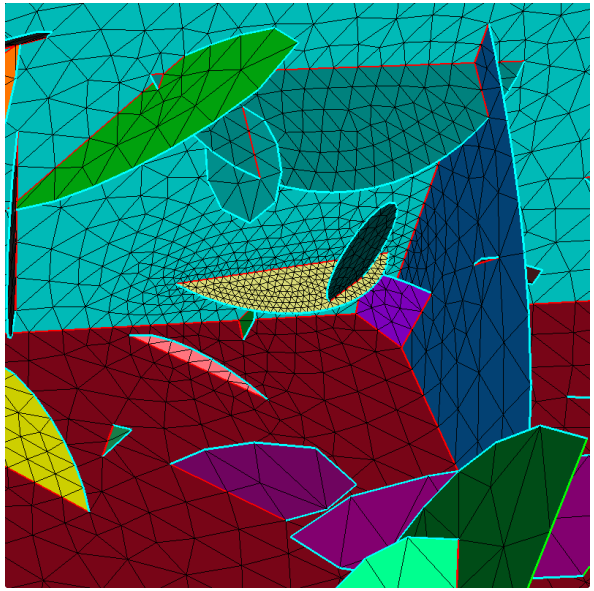
$$p_{k,\gamma} = p_k, \quad \text{on } \Sigma_k, \forall \gamma \in F_k, \text{ with } F_k \text{ the set of fractures with } \Sigma_k \text{ (the k-th intersections) on the boundary,} \quad \textit{(Continuity of the hydraulic heads)}$$

$$\sum_{\gamma \in F_k} \mathbf{u}_{k,\gamma}.\mathbf{n}_{k,\gamma} = 0, \quad \text{on } \Sigma_k. \quad \textit{(Conservation of fluxes)}$$

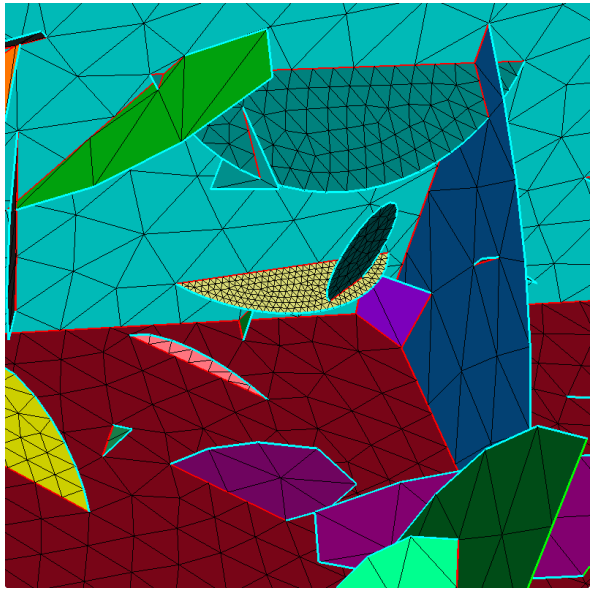
Mesh generation

Mesh generation with MODFRAC software (Inria Gamma & UTT)

Cube size L	# fractures	Time for mesh generation hh:mm:ss	#triangles
100	152,405	00:02:44	2,055,624
150	508,339	00:06:29	12,890,943
200	**1,176,566**	**00:14:37**	20,522,575



Matching surface mesh



Non-matching surface mesh

On a Laptop Intel Core i7
4 cores CPU
32GiB RAM

Parallelism using 4 POSIX-threads

H. Borouchaki, P. Laug, P.L. George, Parametric surface meshing using a combined advancing-front & generalized-Delaunay approach, Int. J. for Numerical Methods in Engineering, 2000

The HHO method

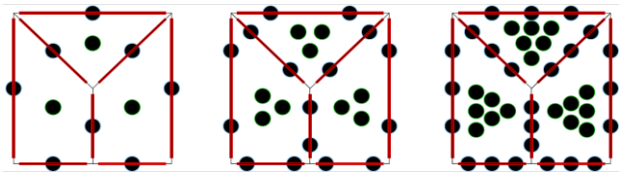
The Hybrid High Order (HHO) method

- Discrete unknowns on each finite element are:
  - On mesh faces
  - Inside the elements
- Algebraically, we obtain a Schur complement system for only the faces unknowns, and known to be symmetric positive definite.
- Local gradient reconstruction from cell and face unknowns
- A stabilization term ensures that the traces of the cell unknows and the face unknowns match in a least-squares sense.
- Conservative fluxes are built from the sum of two terms
  - ✓ the normal component of the reconstructed gradient
  - ✓ A correction term coming from the stabilization
- Main advantages of HHO:
  - ✓ **Feature 1**: Globally and locally **mass conservative method** (whatever $k \geq 0$)
  - ✓ **Feature 2**: Face polynomials of **order $k \geq 0$**
  - ✓ **Feature 3**: Support **general meshes** (polygonal/polyhedral cells)

Di Pietro, D.A and Ern, A. and Lemaire, S. Computational Methods in Applied Mathematics (2014)

Cicuttin, M. and Di Pietro, D.A. and Ern, A. Journal of Computational and Applied Mathematics (2018)

Cockburn, B. and Di Pietro, D.A. and Ern, A. ESAIM: M2AN (2016)



order k=0 order k=1 order k=2

✓ Software:
- ❑ **Library Disk++ for HHO (open-source)**
- ❑ **NEF++ (Inria) for an application to DFNs**
- ❑ **Prune (Inria) for automatic parameter studies**

The Hybrid High Order (HHO) method: basis functions

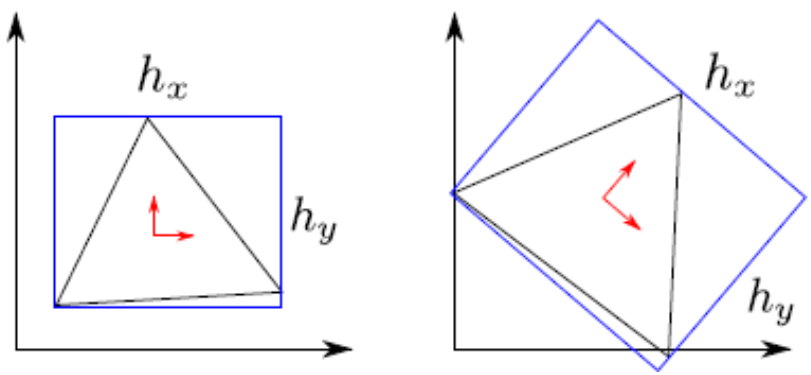
- HHO unknowns are polynomial coefficients (no physical interpretation)
  - On mesh faces: 1, x, x^2, x^3, …
  - Inside the elements : 1, x, y, x^2, y^2, xy, …
- No reference element
- Defined on physical element
- No continuity is required between face and cell

A lot of possibilities
BUT
the choice may affect the condition number of local matrices and performances

- **Hierarchical scaled centered and rotated monomials**

$$\tilde{x} = \frac{(x - x_b)\cdot a_1 + (y - y_b)\cdot a_1}{h_x} \in [-1, 1]$$

$$\tilde{y} = \frac{(x - x_b)\cdot a_2 + (y - y_b)\cdot a_2}{h_y} \in [-1, 1]$$



Advantages

- **Hierarchical** : easy to change order *k* (only few monomials to compute)
- **Centered** : independent of translation (use barycenter)
- **Scaled** : independent of homothety (use length of the bounding box)
- **Rotated** : independent of rotation (use inertia axes)

The HHO method in DFN flow simulations

Feature 1 of HHO: mass conservative method (whatever k ≥0)

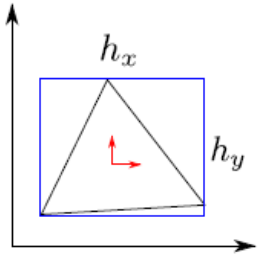
	k	$\lvert Q_{in,x} + Q_{out,x} \rvert$ Cartesian	Rotated
L100	0	**1.44e-07**	**9.82e-11**
	1	**1.66e-4**	**5.26e-10**
	2	**2.31e-2**	**4.63e-09**
	3	**15.21**	**4.20e-08**
L150	0	**3.12E-07**	**7.32e-10**
	1	**1.58e-3**	**1.31e-08**
	2	**38.22**	**1.85e-08**
	3	**45.73**	**4.85e-08**

- **Choice of the basis functions**

✓ **Hierarchical scaled centered and rotated monomials (Rotated)**

$$\tilde{x} = \frac{(x - x_b)\cdot a_1 + (y - y_b)\cdot a_1}{h_x}$$

$$\tilde{y} = \frac{(x - x_b)\cdot a_2 + (y - y_b)\cdot a_2}{h_y}$$

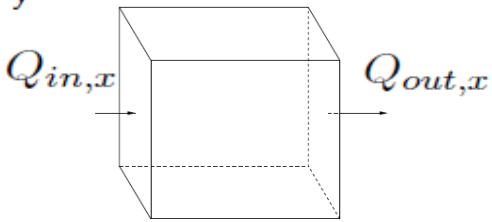


× **Hierarchical scaled centered barycentric monomials (Cartesian)**

$$\hat{x} = \frac{(x - x_b)}{2h_x}$$

$$\hat{y} = \frac{(y - y_b)}{2h_y}$$

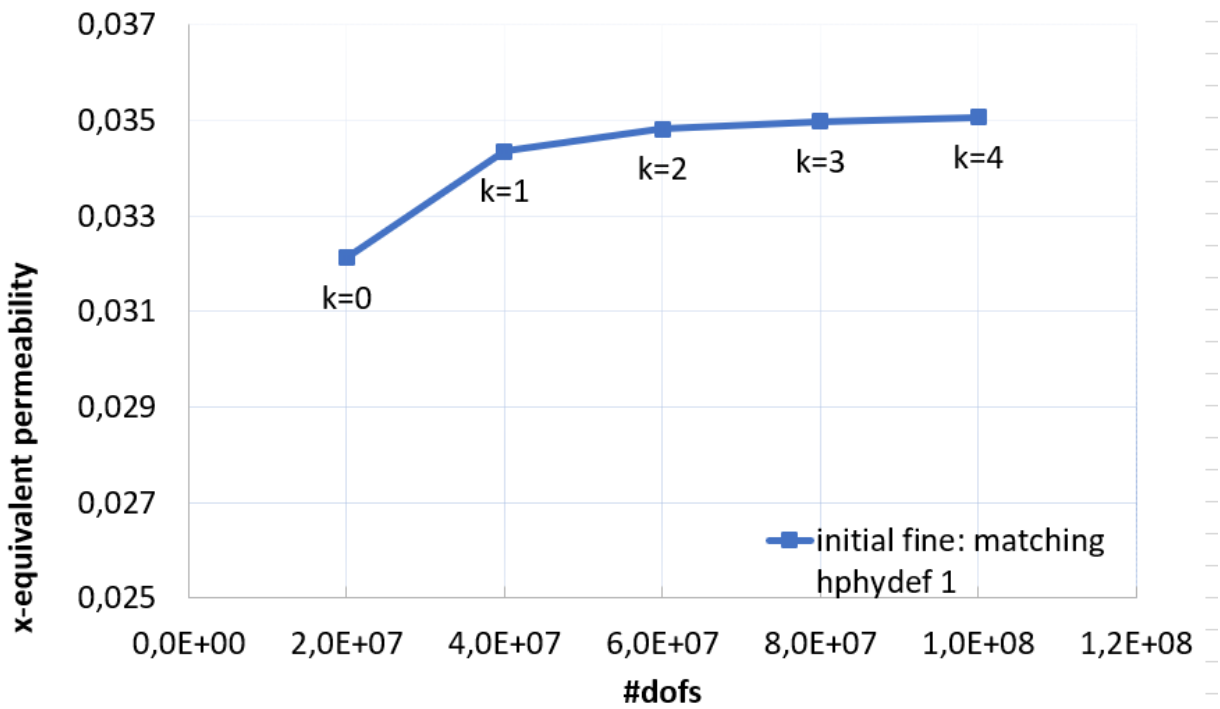
- **Hierarchical scaled centered and rotated monomials**
  - ✓ **Improve the condition number of the local matrices**
  - ✓ **Global and local mass conservation ensured whatever k**



Feature 2 of HHO: high order

- ❑ Estimation of the *x*-equivalent permeability:
- ✓ Test case L150: **508,339 fractures**
- ✓ **Fine mesh: 12,890,943 triangles**
- ✓ Estimation K_x **= 3.51e-2 m²·s⁻¹**

$$K_x = \frac{Q_{in,x}}{L\,\Delta h}$$

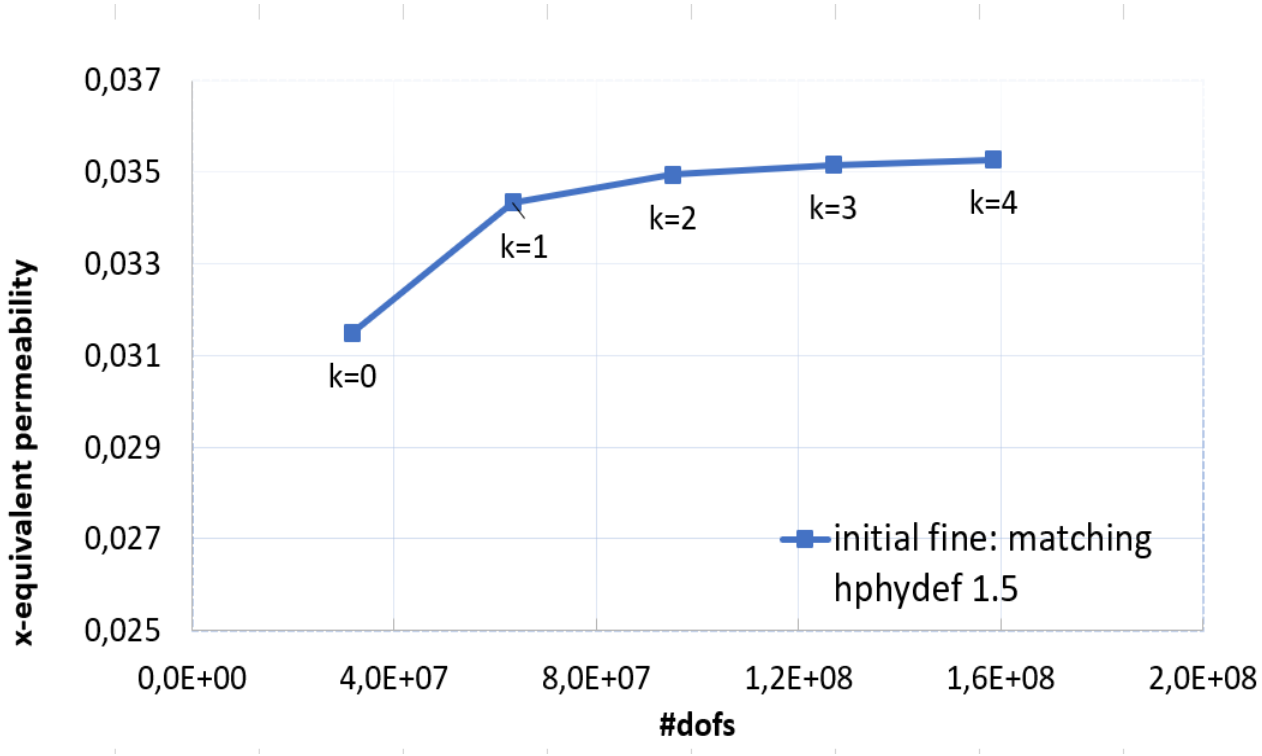


Inria Cluster CLEPS: 4x Intel Xeon E7-4860 v2; 12 cores, 2.6-3.2GHz; **3T RAM Memory**

Feature 2 of HHO: high order

- ❑ Estimation of the x-equivalent permeability:
- ✓ Test case L200: **1,176,566 fractures**
- ✓ **Fine mesh: 20,522,575 triangles**
- ✓ Estimation K_x **= 3.53e-2 m^2.s^{-1}**

$$K_x = \frac{Q_{in,x}}{L\,\Delta h}$$

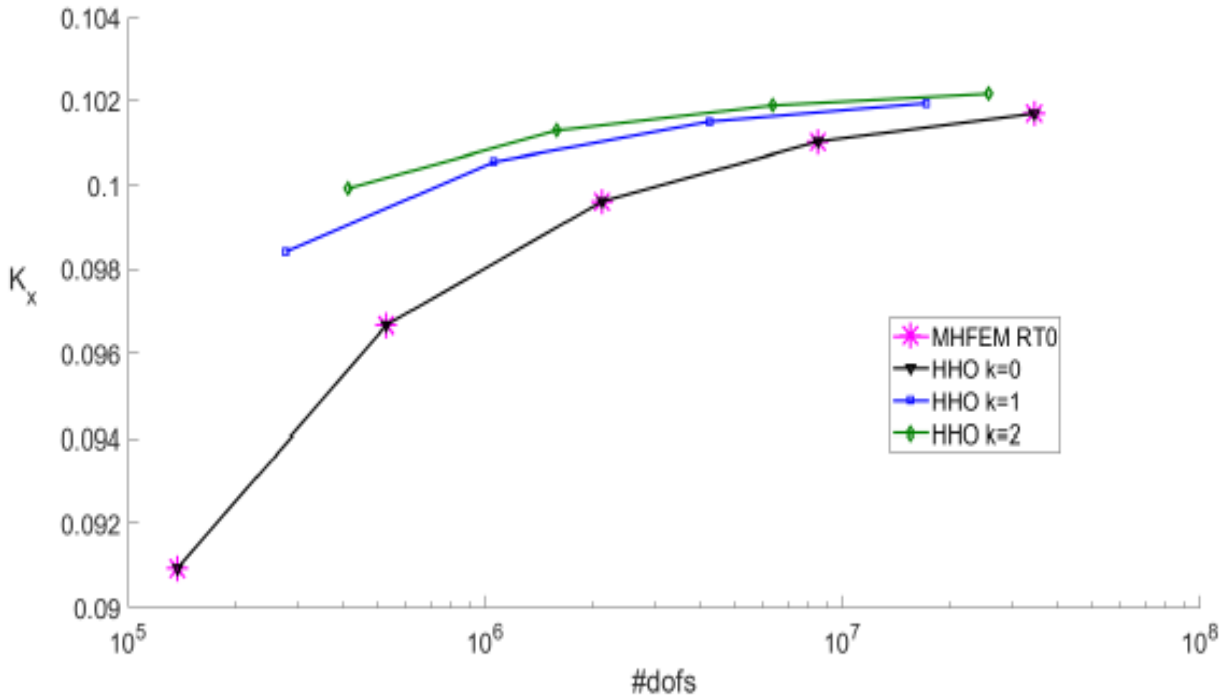


Inria Cluster CLEPS: 4x Intel Xeon E7-4860 v2; 12 cores, 2.6-3.2GHz; **3T RAM Memory**

Feature 2 of HHO: high order

- ❑ Benefit of **high order vs mesh refinement**
- ✓ Small test case: **1,397 fractures**

Hédin, F., Pichot, G., Ern, A. Numerical Mathematics and Advanced Applications ENUMATH 2019



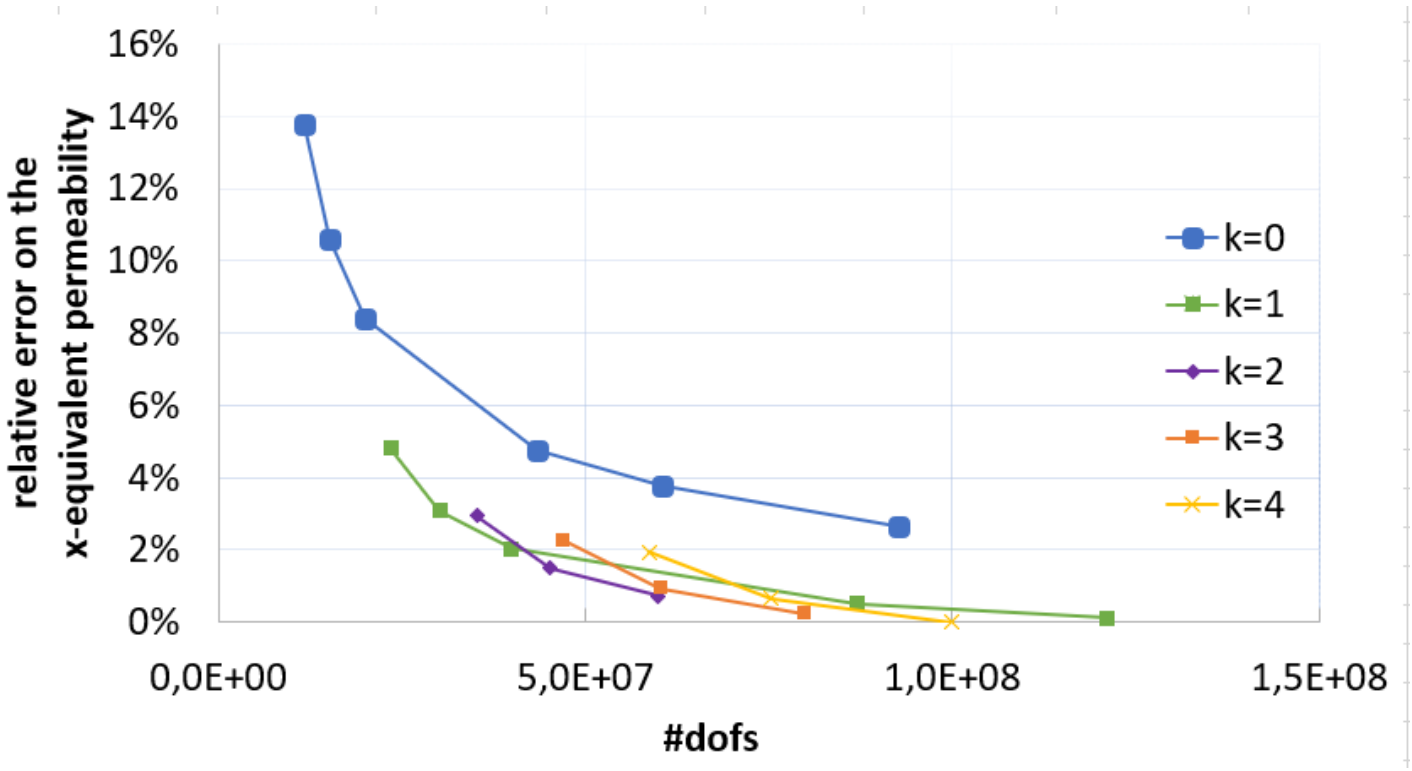
$$K_x = \frac{Q_{in,x}}{L\,\Delta h}$$

- ✓ nE the total number of edges
- ✓ For k, the number of dofs are (k+1) nE

- ✓ For a fixed #dofs, the equivalent permeability is **better approximated with k>0**

Feature 2 of HHO: high order

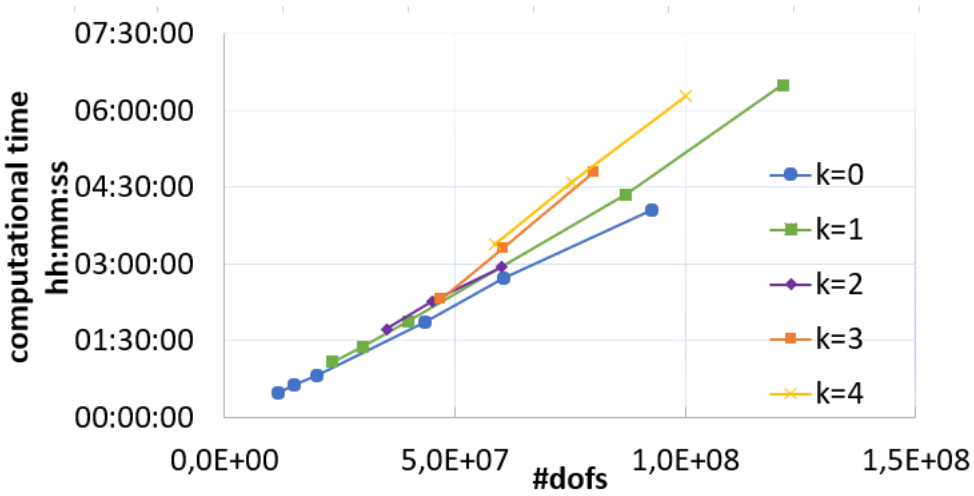
- ❑ Benefit of **high order vs mesh refinement**
- ✓ **L150: 508,339 fractures**

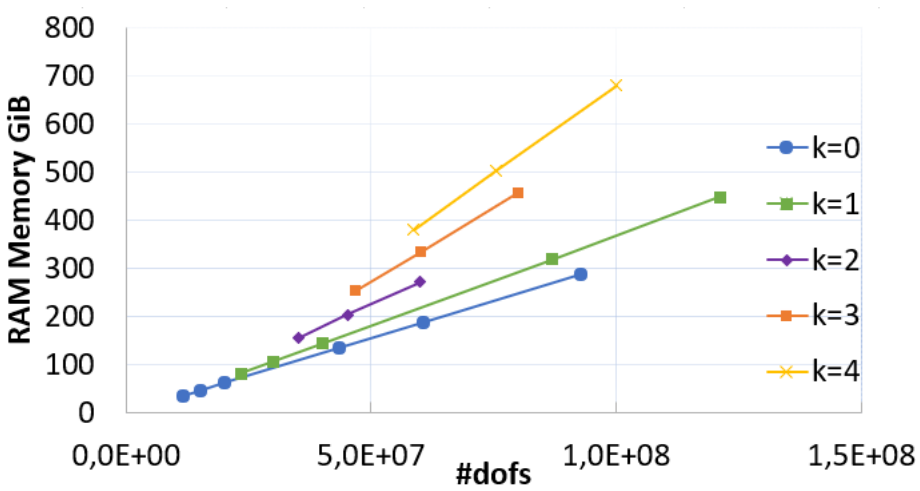


- ✓ **On a coarse mesh, high order >1 is advised**
- ✓ **On a fine mesh, order 1 is enough**

Feature 2 of HHO: high order

- Benefit of **high order vs mesh refinement : computational time & RAM**
- ✓ **L150: 508,339 fractures**





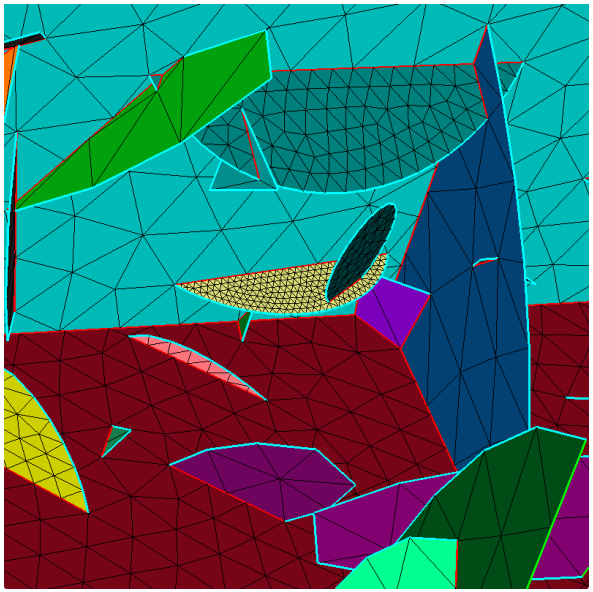
- ✓ **For a given #dofs, the extra-time of high order is negligible wrt order 0 for $k \leq 2$**
- ✓ **For a given #dofs, the RAM memory requirement increases with *k***

k	#dofs	Time	RAM GiB
0	43,531,537	01:51:14	133
1	39,998,460	01:52:24	144
2	45,222,504	02:15:48	203
3	46,930,868	02:18:27	253

Feature 3 of HHO: polygonal/polyhedral cells

- **Objective:** Use this feature to save computational resources

- **How ?**
  - **With non matching mesh:** the mesh generator **MODFRAC** (Inria&UTT) allows to choose an independent mesh step from one fracture to another
  - **With the channelling effect of DFN flow:** the fractures that carry most of the flow could be refined, the other could be coarsened.
  - **With the creation of polygons** from the non-matching discretization



Non-matching surface mesh

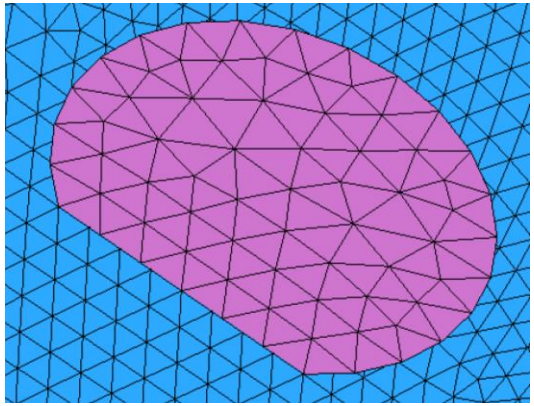
- **Limitation:** Coarsening is constrained by the geometry and intersections of the fractures!

- Q1: How to create the polygons from the non-matching meshes?
- Q2: How to select the set of fractures that needs to be refined ?

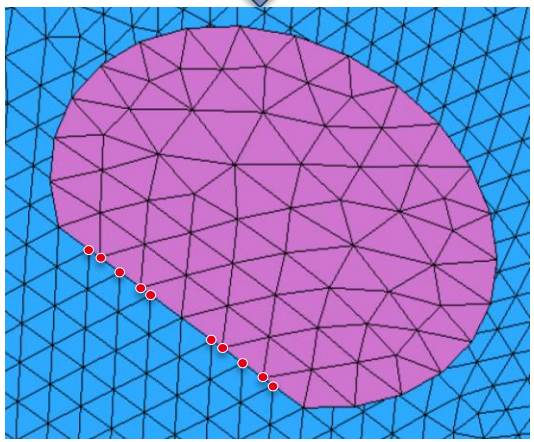
Feature 3 of HHO: polygonal/polyhedral cells

To address Q1: create the polygons from the non-matching meshes

- **Development of a node insertion algorithm**

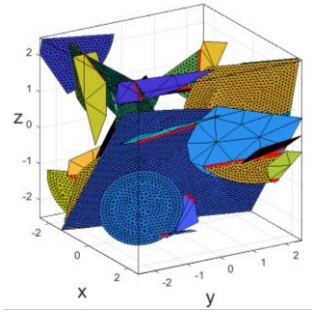


non-matching triangular mesh

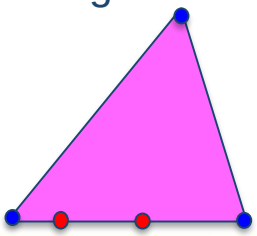


polygonal mesh

- **Algorithm to create the polygons:**
  1. Load the non-matching triangular mesh, especially the edges of each intersection on both side
  2. Define the linear coordinates of the vertices on the intersection to **find where to insert the vertices**
  3. Define a **new global numbering** of the intersection vertices (no duplicate indices!)
  4. Add the extra-vertices to **create the polygons** (red dots) and check the insertion in the connectivity table that defines the polygons in terms of vertices
  5. Compute the **2D coordinates** of extra vertices in the fractures they now belong to



- **Challenges**
  - Large number of intersections
  - Efficient algorithms

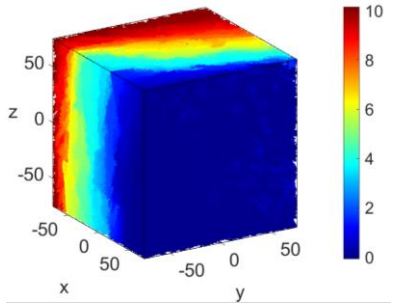


- Vertices of the triangles
- Extra vertices to create a polygon

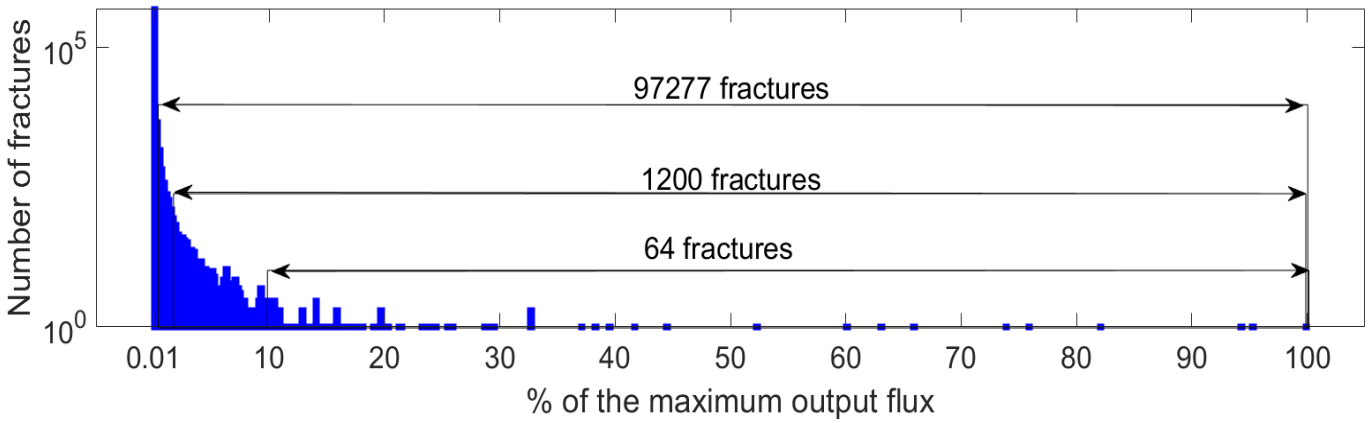
Feature 3 of HHO: polygonal/polyhedral cells

To address Q2: select the set of fractures that needs to be refined :

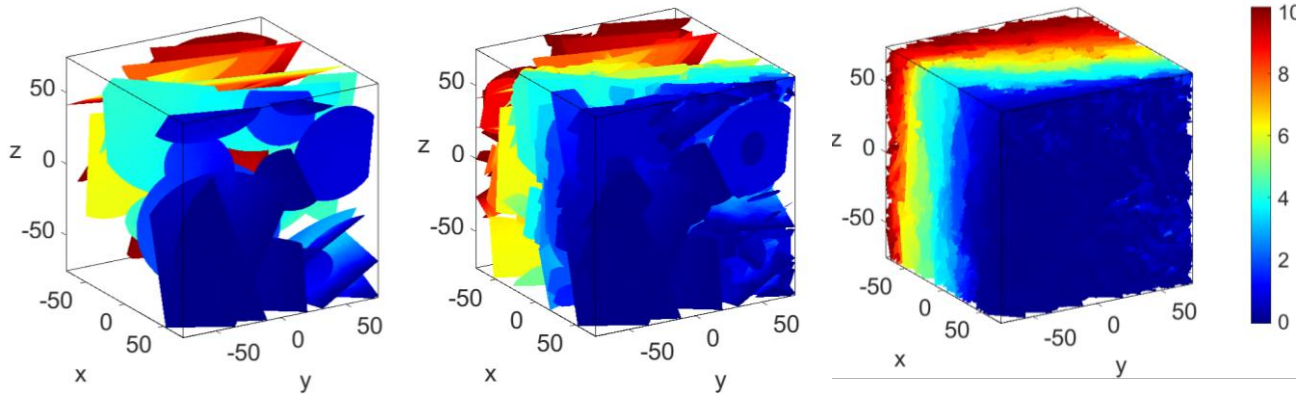
- **A first basic idea:** Compute the output flux per fracture **from a former coarse flow simulation**



L150: **508,339** fractures
x direction: h1:10, h2 :0
Mesh load: 6 min,
Computation: 16 minutes (Laptop)
Coarse mesh: 7,231,104 triangles



Threshold: % of the maximum output flux	#fractures above the threshold
10%	64
1%	1200
0.01%	97,277

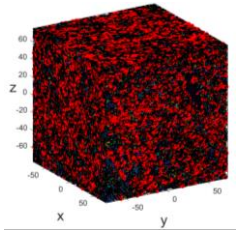


64
(0.01% of the DFN)

1200
(0.23% of the DFN)

97,277
(19% of the DFN)

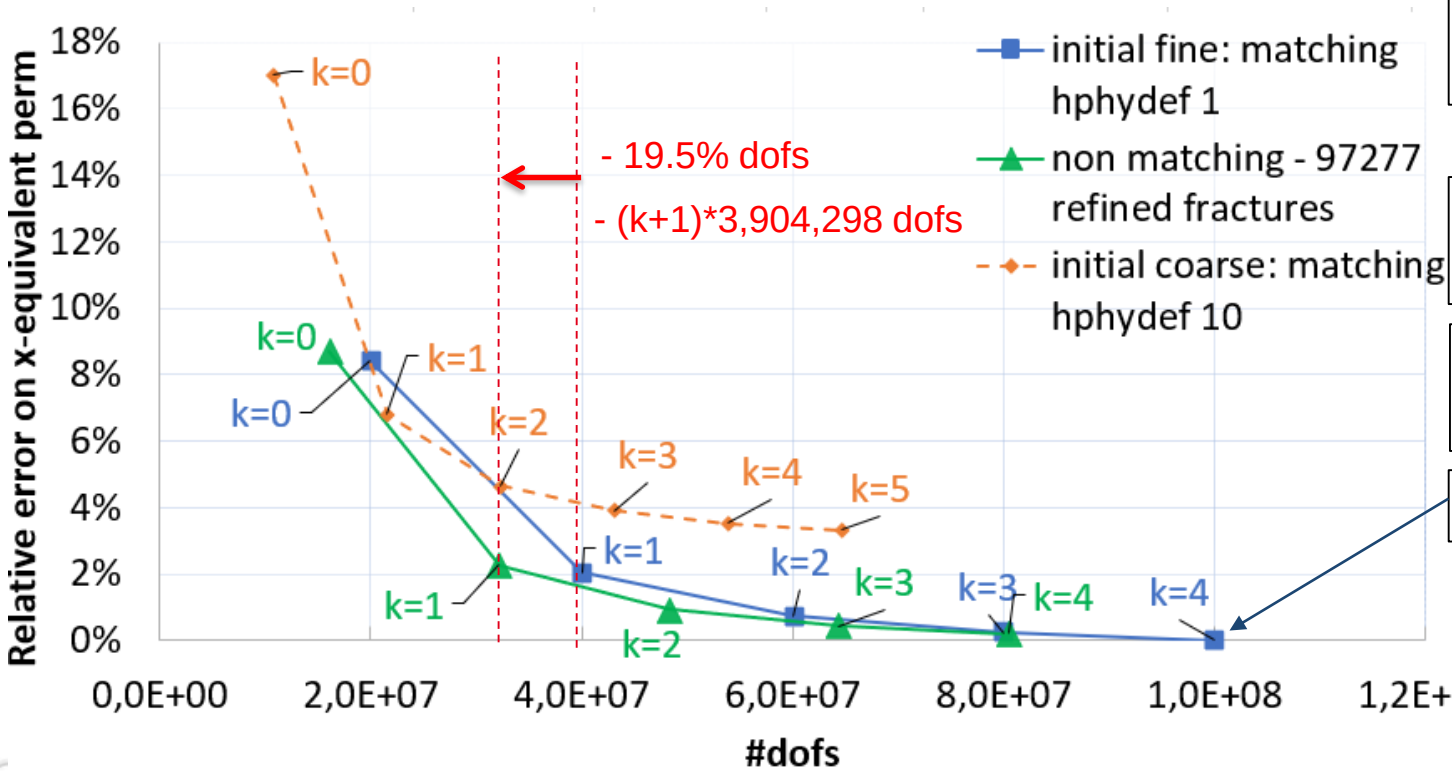
Feature 3 of HHO: polygonal/polyhedral cells



- **POLYGONAL mesh**

Thanks to the coarse simulation, we keep a mesh step of 1 for 97,277 selected fractures (19% of the DFN) and coarsen the other fractures (mesh step of 2)

L	# fractures	#tri	#extra-vertices (red dots)	%polygons	#tri wrt fine matching mesh
150	508,339	10,461,407	201,583	3.5%	**-19%**



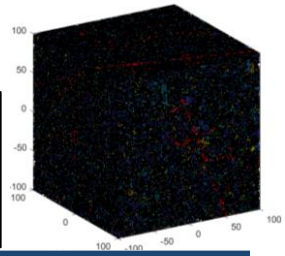
-15% time
-12% to -15% RAM

Initial fine mesh: 12,890,943 triangles

Initial coarse mesh: 7,029,255 triangles

K_x = 3.51e-2 m^2.s^{-1}

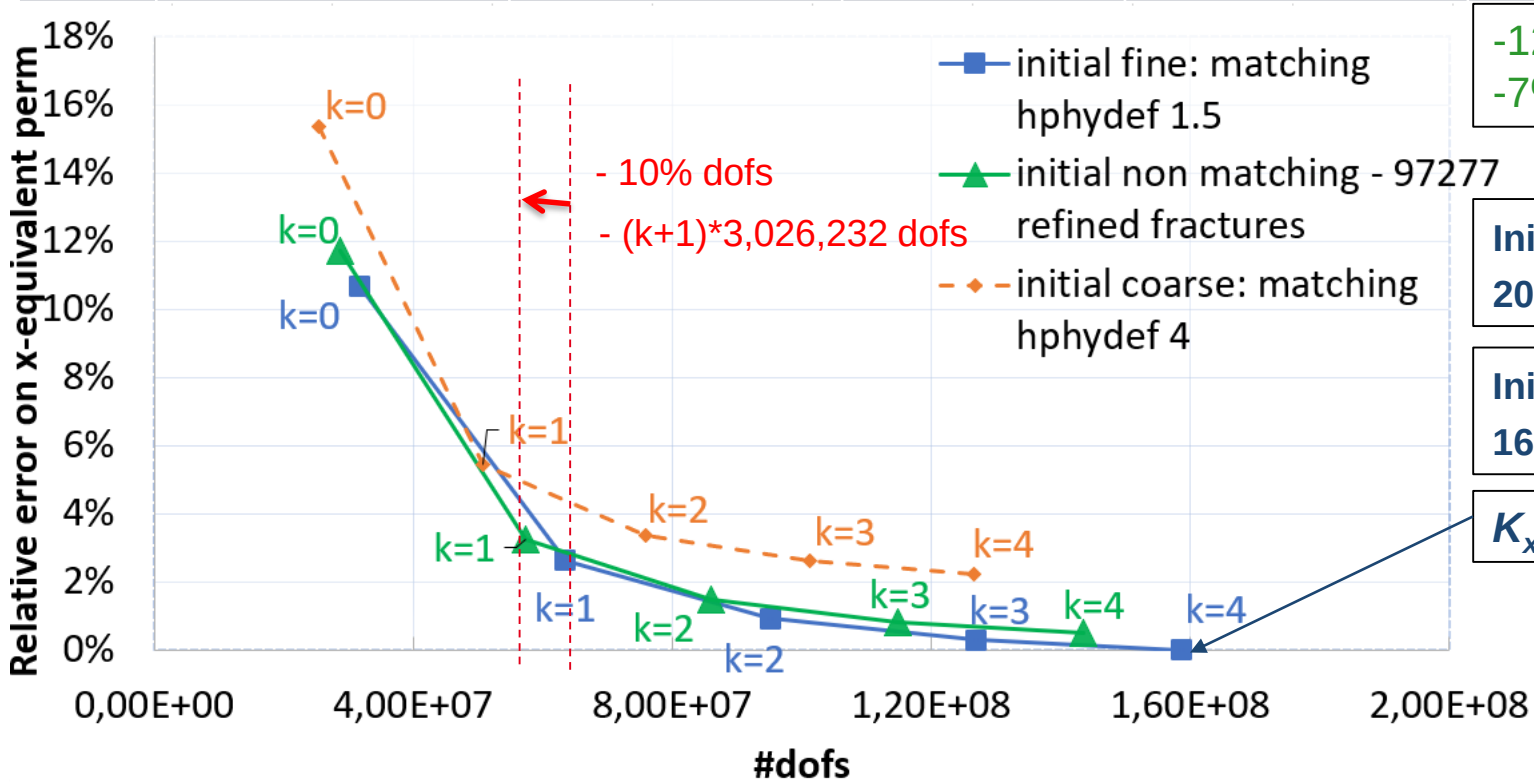
Feature 3 of HHO: polygonal/polyhedral cells



❑ POLYGONAL mesh

To save a coarse simulation, we keep the 97,277 selected fractures (8% of the DFN) of test case L150 and mesh them with a mesh step of 1.5, the others with a mesh step of 2.

L	# fractures	#tri	#extra-vertices (red dots)	%polygons	#tri wrt fine matching mesh
200	1,176,566	18,648,084	96,143	1.0%	**-9%**



-12% to -16% time
-7% to -10% RAM

Initial fine mesh: 20,522,575 triangles

Initial coarse mesh: 16,554,413 triangles

K_x = 3.53e-2 m^2.s^{-1}

Conlusion

- We have **successfully** tested the **HHO method on large scale DFN** (more than **1 million of fractures**)
- The implementation of the method is **locally and globally conservative** (whatever k ≥0) thanks to the use of basis functions based on **hierarchical scaled centered and rotated monomials**
- The use of high order is an advantage to compute a **more accurate** flow in DFN **without much extra time compared to a low order method** but the **RAM memory demand increases with *k* for the same #dofs.**
- We have shown that the use of non-matching meshes, the channelling effect of DFN flow and the polygonal feature of the **HHO method allows to reduce the number of #dofs, and therefore the time computation and RAM requirements** in DFN flow simulations.

Perpectives

- Further reduction of the #dofs by using the **additional recent features of HHO**
  - ✓ **Unfitted HHO**
  - ✓ Possibility to **merge the triangles** in (possibly non convex) **polygons**

- A **C-version** of the **node insertion algorithm**

- Studying **an iterative solver to reduce the RAM memory requirements**

- Using a **cheaper algorithm to know which fractures to refine :**
  - ✓ From **graph connectivity of the fracture network** and BCs.
  - ✓ No need any flow computation

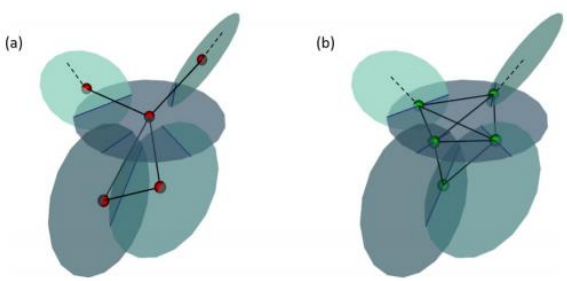


FIG. 1. DFNs and equivalent graphs: (a) fracture graph, (b) intersection graph. ©From Doolaeghe et al. 2020

D. Doolaeghe & P. Davy & J. Hyman & C. Darcel. Graph-based flow modeling approach adapted to multiscale discrete fracture network models (2020), Physical review E, 102(5), 053312.

E. Burman and A. Ern, An unfitted hybrid high-order method for elliptic interface problems, SIAM J. Numer. Anal., 56(3), 1525-1546 (2018)

A. Miraçi, J. Papež, and M. Vohralík (2020)
A Multilevel Algebraic Error Estimator and the Corresponding Iterative Solver with *p*-Robust Behavior. SIAM J. Numer. Anal., 58(5), 2856–2884.

M.Vohralík (2007). A Posteriori Error Estimates for Lowest-Order Mixed Finite Element Discretizations of Convection-Diffusion-Reaction Equations. SIAM Journal on Numerical Analysis. 45.

S. Karra, D. O'Malley, J. Hyman, H. Viswanathan and G. Srinivasan (2017). Modeling flow and transport in fracture networks using graphs. Physical Review E. 97.

Thanks a lot for your attention !