VI International Conference on Adaptive Modeling and Simulation
ADMOS 2013
J. P. Moitinho de Almeida, P. Díez, C. Tiago and N. Parés (Eds)

CONFORMAL HEXAEDRAL MESHES AND ADAPTIVE MESH REFINEMENT

GÉRALD NICOLAS*, THIERRY FOUQUET*

* EDF R&D
1 avenue du Général de Gaulle
92140, Clamart, France
e-mail: gerald.nicolas@edf.fr
e-mail: thierry.fouquet@edf.fr

Key words: h-refinement; adaptive meshing; hexahedron; conformal mesh; finite element; structural mechanics

Abstract. During a numeric simulation based on the finite element method, the h-refinement of the mesh consists in splitting the elements where an error indicator is higher than a threshold. One major point is that the final mesh must be conformal. When the mesh is defined only by triangles or tetrahedra, the junction between two zones with a different level of refinement has been solved for many years. When the mesh is made of hexahedra, this junction cannot be made of hexahedra. A proposal is made in this paper to connect the zones with some specific elements. Two applications are presented here and show the efficiency of the method.

1. INTRODUCTION

In a numerical simulation using the finite element method, the mesh has to be fine enough to guarantee the accuracy of the solution. To achieve this goal, mesh adaptation offers an effective compromise, combining a fine mesh with a low computational cost. When using the h-refinement method, some meshes are divided but difficulties occur at the interface between two zones with different levels of refinement, if a conformal mesh is required. That problem is solved either by specific finite elements in the junction [1] or by a specific splitting of these meshes [2].

If the initial mesh is made of triangles or tetrahedra, the splitting of the meshes at the interface produces new triangles or tetrahedra. Since the early 90's, this method has been implemented in HOMARD, our software for mesh refinement ([3], available in [4]). But in some numeric simulations, the initial mesh is made of hexahedra because they are more efficient than the tetrahedra. In that case, the transition is not as simple as it is with the tetrahedra: the conformal connection cannot be made with others hexahedra. To solve this problem, we developed a new method. First, the error indicator from the computed solution is used to produce a non-conformal mesh [5]. Then, every hexahedron that is located at the interface between the zones of different levels of refinement is examined: using tetrahedra and pyramids makes possible a conformal connection.

The first part of this paper, chapter 2, presents a basic application on a 3D structure that shows the advantages of the h-refinement when the mesh is made of hexahedra. The transition zone will be described and the convergence of the computation is easily reached. Secondly, chapter 3, the central part of the method is detailed. Last, chapter 4, an industrial application of the method is presented.

2. FIRST EXAMPLE OF APPLICATION

To give an idea of the effectiveness of the h-refinement with a mesh made of hexahedra, we present an example in structural mechanics. Lo et al suggested this test case in [1]. The model represents a 3D cross with one fixed face (see. Figure 1). A uniform pressure is applied on one face. The objective is to get an accurate value of the field of displacement along the line opposite to the fixed face. The simulations are done with *Code_Aster* [6], the open source finite element software for mechanical analysis. The mesh adaptations for this test case have been driven using the goal-oriented estimation of the error [7]. The goal of the mesh adaptation is to increase the accuracy on the displacement.

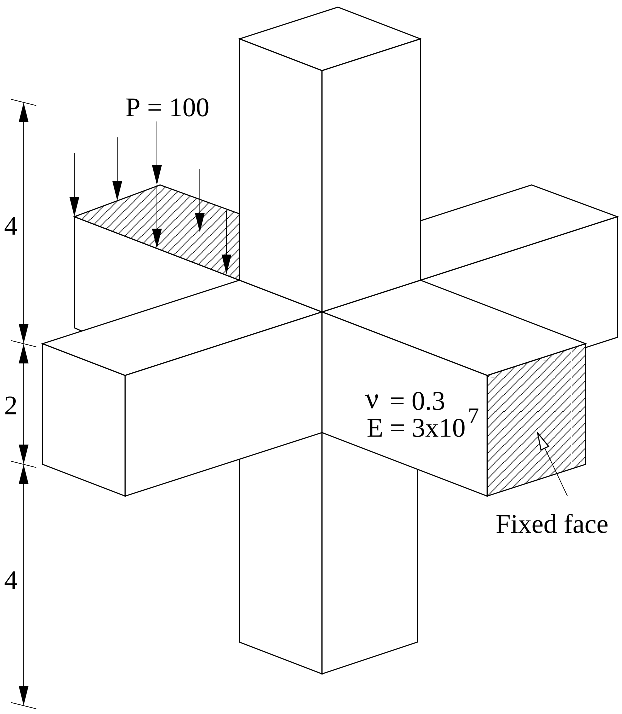


Figure 1: Description of the 3D cross

The initial mesh is composed of second order hexahedra. The threshold is established as 10% of the largest error in the initial mesh. The adaptation stops when the global error indicator on the structure is 20% of its value on the initial mesh. The evolution of the refined meshes along the iterations of the adaptation ensures the diminution of the global error of the problem. When the convergence is reached, the aspect of the meshes all along the iterations is similar to those obtained by Lo. When we look at the meshes (see. Figure 2), we see that the refinement occurs near the fixed face and near the edges between two branches of the cross, as expected. The transition zones between two different levels of refinement are visible in the figure each time a volume is not a hexahedron anymore.

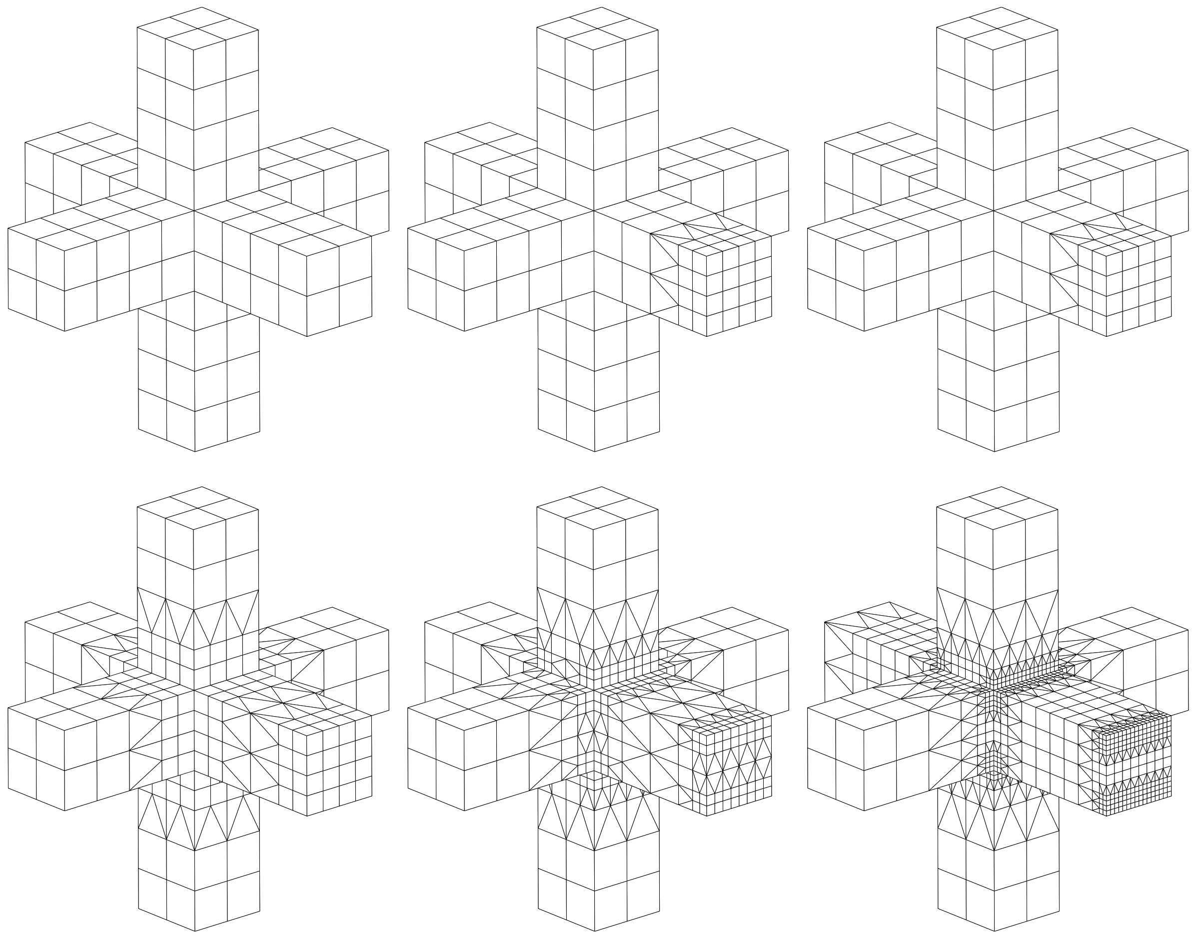


Figure 2: Initial and adapted meshes of the 3D cross

For some different strategies (see. Figure 3), we present the value of the total error plotted against the number of nodes, in a log-log-plot. First, a uniform refinement is tested (circle). As expected, a fast convergence cannot be reached with a uniform refinement. In the other cases, the threshold is defined either by an absolute value (abs plot), or by the percentage of the "worst" meshes (pcm plot). We can see that the influence of the strategy on the speed of convergence is rather small. Whatever the choice, the convergence is reached with more or less the same number of degrees of freedom. This conclusion is coherent with our experience: the most important is to catch the elements where the error is high and to be sure that they are refined.

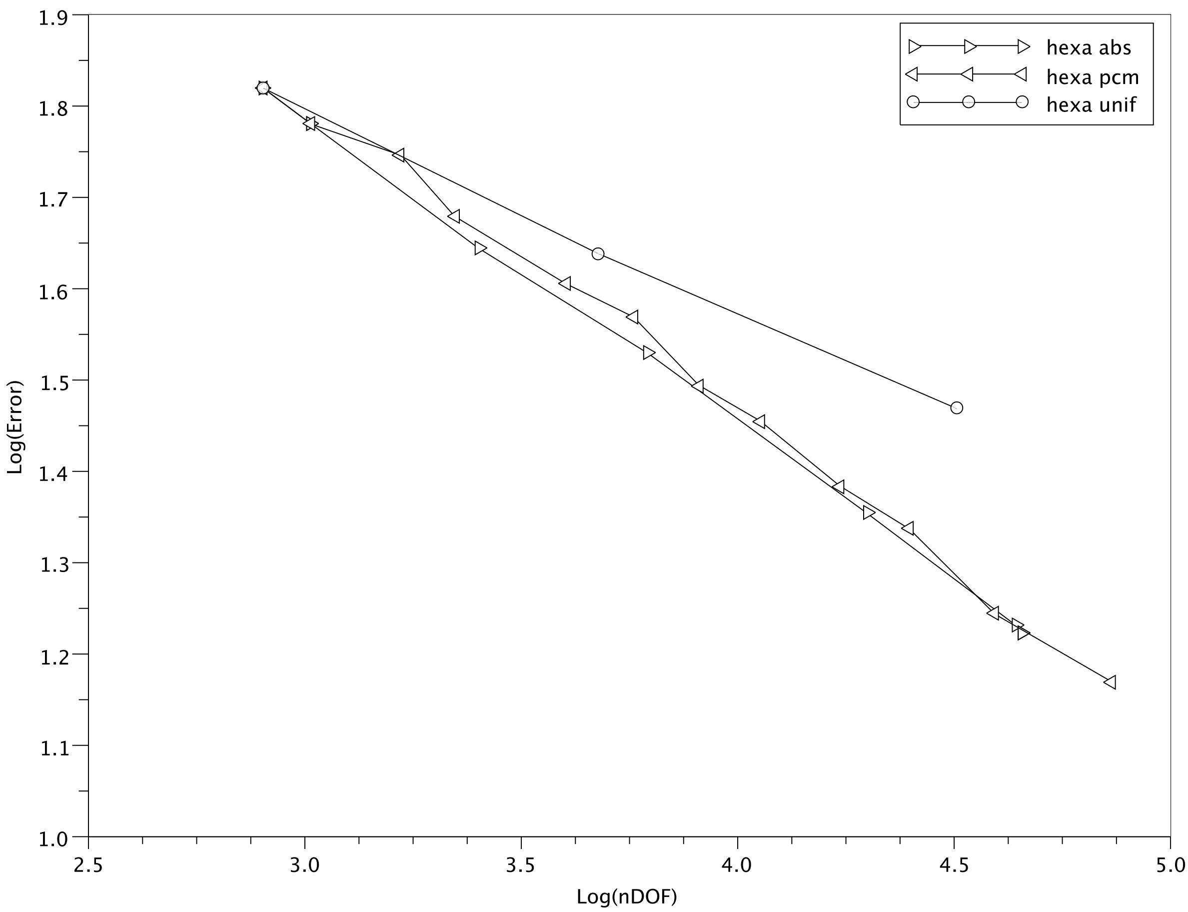


Figure 3: Convergence

3. ALGORITHM

The algorithm is described with details in [3]. The guidelines are presented here; a special focus is made on the technique that is employed to produce a conformal mesh when hexahedra are present.

3.1. Basic ideas

At the beginning of the process, all the elements of the initial mesh belong to level #0. The computation of the physical problem produces an error indicator over every element. Giving a threshold identifies some elements: those where the error indicator is higher than this threshold are split. That phase creates some elements that belong to level #1. Then, we have to solve the junction between the two different levels of refinement by introducing some special divisions of the elements to produce a conformal mesh. A new computation is made over this new mesh and the same adaptation can be processed one more time, until convergence is reached.

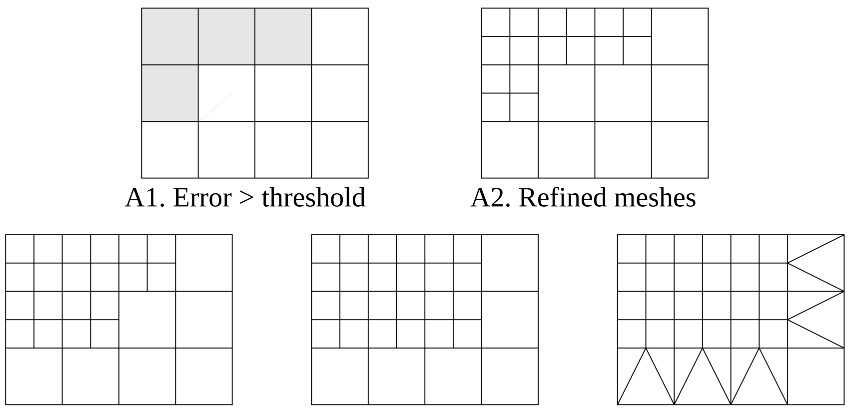
In our method, we decided to deal with 3D meshes and 2D meshes as well. When the mesh is composed of 3D elements, the faces of every element are defined in the data structure. If a

3D element has to be split because of the value of the error indicator, the decision of refinement is transferred to its faces. Then the resolution of the conflicts between the levels of refinement is made uniquely by considering the 2D faces. This is the central part of the algorithm and it is the same whatever the composition of the initial mesh. Last, examining the final decisions of its faces makes the refinement of every 3D element.

We illustrate this technique for a 2D mesh made of quadrangles. It is based on a three-stage procedure:

A. Mark every quadrangle that requires refinement according to the error indicator
B. Mark every quadrangle that has at least two refined neighbours
C. Stop the propagation using transition elements

These stages are illustrated in Figure 4. Suppose that the grey quadrangles are the elements over which the error is higher than the threshold (A1). The first action of the phase A is to refine these elements: all their edges are equally split in two by placing a node at their midsections, so that four internal quadrangles are created (A2). The central quadrangle with two refined neighbours is then refined (B1). In our example, that action modifies the status of its neighbour on the right: it now has two split neighbours so it also has to be refined (B2). At this point, there are no more quadrangles with 2 refined neighbours or more. In the last step, all the quadrangles with one refined neighbour are split using a special technique (C).



A1. Error > threshold   A2. Refined meshes

B1/B2 Rules of 2 neighbours   C. The transition meshes

Figure 4: Algorithm for a 2D mesh

3.2. The transition elements

A special point must be made regarding the transition elements. They are defined to ensure a conformal transition between two different levels of refinement. When 3D elements are present in the mesh, at the end of the phase B of the algorithm, these 3D elements can be sorted by the status of their faces:

- No face is split: the 3D element is kept as is
- All the faces are regularly split: the 3D element is regularly refined
- Other situations: the 3D element is located into a transition zone between two different levels of refinement. Each situation must be analyzed (see. 3.2.2)

3.2.1. Refinement of a quadrangle

At the end of the phase B of our algorithm (Mark every quadrangle that has at least two refined neighbours) every quadrangle is in one of these 3 situations: either no edge is split, or a unique edge is split, or all edges are split. If all the edges are split, placing a node at its midsection equally splits each edge. Adding a node at the centre of the quadrangle produces the 4 quadrangles of the regularly refinement

When a unique edge is split, some transition elements are needed. We decided that no additional node on any edge would be introduced to create them. Consequently, the pending node is connected to the opposite vertex, to produce three triangles.

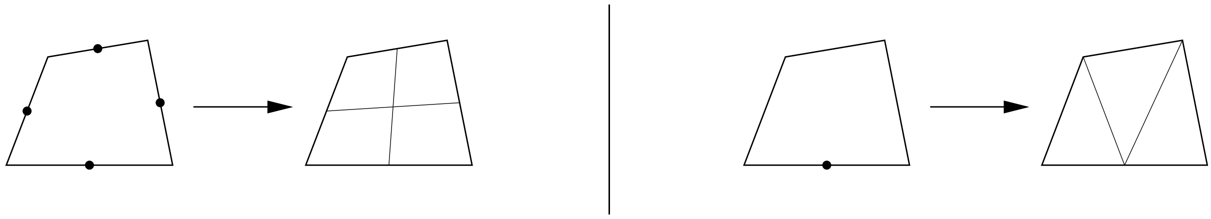


Figure 5: Refinement of a quadrangle: regular and transition

3.2.2. Transition elements for the hexahedra

The situation is much more complex when a hexahedron is located at the interface between two zones with a different level of refinement. The analysis of this case constitutes the central point of our work. When the rules are applied, one of six cases can happen at the end of the phase B:

- No edge is split: the hexahedron is kept as is
- All the edges are split: the hexahedron is regularly refined.
- A unique edge is split. The two quadrangular faces that share this edge are refined by transition while the four others faces are kept intact. Two internal edges are created from the middle of the split edge to the two vertices on the opposite edge. This internal division produces four pyramids whose bases are the four non-split quadrangular faces of the hexahedron.

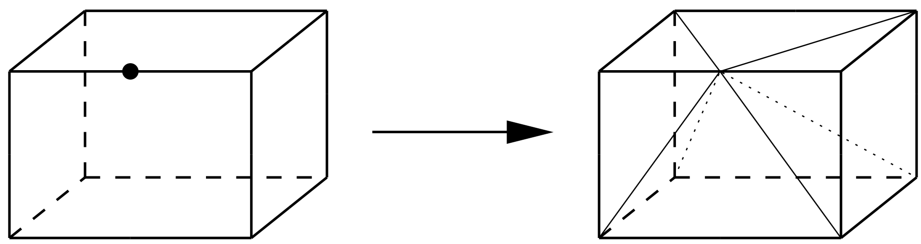


Figure 6: One edge is split

- Two edges on two different faces are split. The two faces of the hexahedron that do not have a split edge are kept intact, while the other four are split into triangles. An additional node is created at the centroid of the hexahedron. Ten internal edges are created with that centroid as a vertex: two are created to connect it to the centre of the two split edges while the other eight are created with the eight vertices of the hexahedron. Two pyramids are then built with bases on the two intact quadrangular faces of the hexahedron, and twelve tetrahedra are created on the remaining triangles.

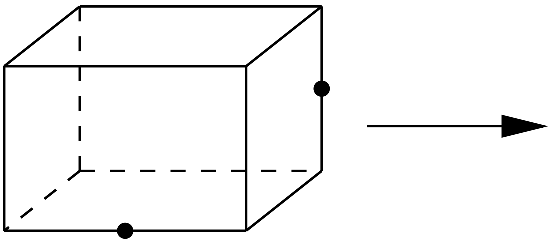


Figure 7: Two edges are split

- Three edges on three different faces are split. Two of these edges cannot belong to the same face, so the six faces of the hexahedron are split in triangles. An additional node is created at the centroid of the hexahedron. Eleven internal edges are created with that centroid as a vertex: three in connection with the centre of the three split edges, while the other eight with the eight vertices of the hexahedron. Eighteen tetrahedra are then created with the eighteen triangles

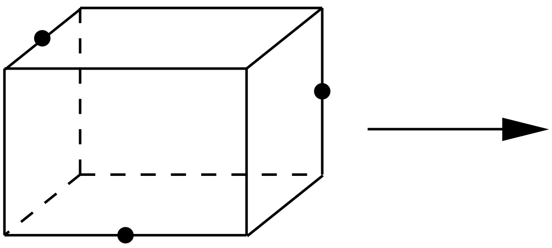


Figure 8: Three edges are split

- A face and its four edges are split. The opposite face of the hexahedron is kept intact, while the other four are split into triangles. Four internal edges are created from the centre of the regularly split face to the vertices of the opposite quadrangular face. A pyramid is then built on the intact quadrangular face of the hexahedron. Four more pyramids are also built on the four quadrangular faces produced by the refinement of the refined face. Last, four tetrahedra are created on the four remaining triangles that are located in the centre of the lateral faces.

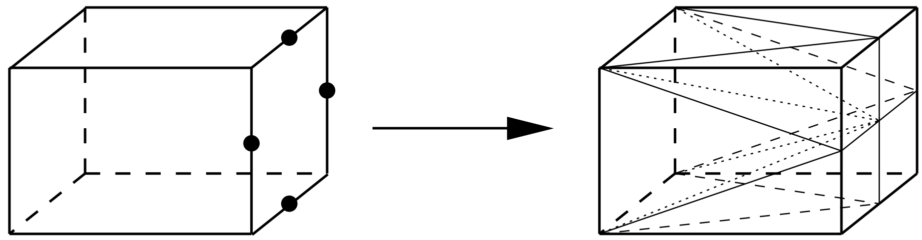


Figure 9: Four edges are split

3.2.3. Two comments about the pyramids

Usually, the hexahedra or the tetrahedra are preferred in a numerical simulation rather than the pyramids because of their properties. However, the pyramidal element offers an effective solution to maintain the compatibility between zones [8]. For the h-refinement, thanks to these pyramidal elements, there is no need to define interfacial relations. The algorithm is the same over the whole mesh, whatever the type of elements.

In a further iteration of the adaptive process, if nothing special is done, the pyramids could be split. There should be a risk in the quality of the mesh. To avoid that phenomenon, we give a temporary status to the transition elements. At the beginning of a new iteration, these transition elements are removed. The algorithm is applied over the plain elements. At the end, new transition elements are added to make a conformal mesh. Doing that, the transition elements are never split and the quality of the mesh is saved.

4. NUMERICAL APPLICATION

We illustrate this algorithm with the simulation of an industrial installation: the mechanical analysis of an arch dam, during the filling up. Two major parts are described: the foundation and the arch. The foundation represents the rocky part around the river valley and gives the stability to the structure. The arch made of concrete is modelled and the variation of the thickness is taken into account.

The initial mesh is mainly made of hexahedra, with a few prisms and tetrahedra in the conformal connection between the arch and the foundation (see Table 1 and Figure 11). The simulation is divided into two parts. Firstly, a calculation is done without any water: the objective is to get an initial state of the stresses in the calculation, considering its weight and the reaction of the foundation. Secondly, the level of the water rises in the upstream side, up to the top of the arch (see Figure 10). The pressure of the water on the upstream face of the arch modifies the field of the stresses and of the displacement in the arch.

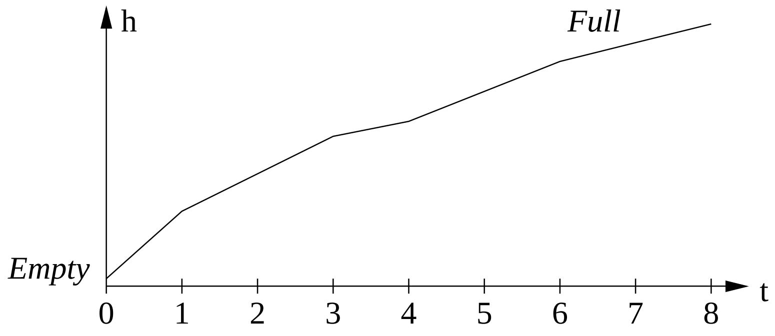


Figure 10 - Filling up of the dam

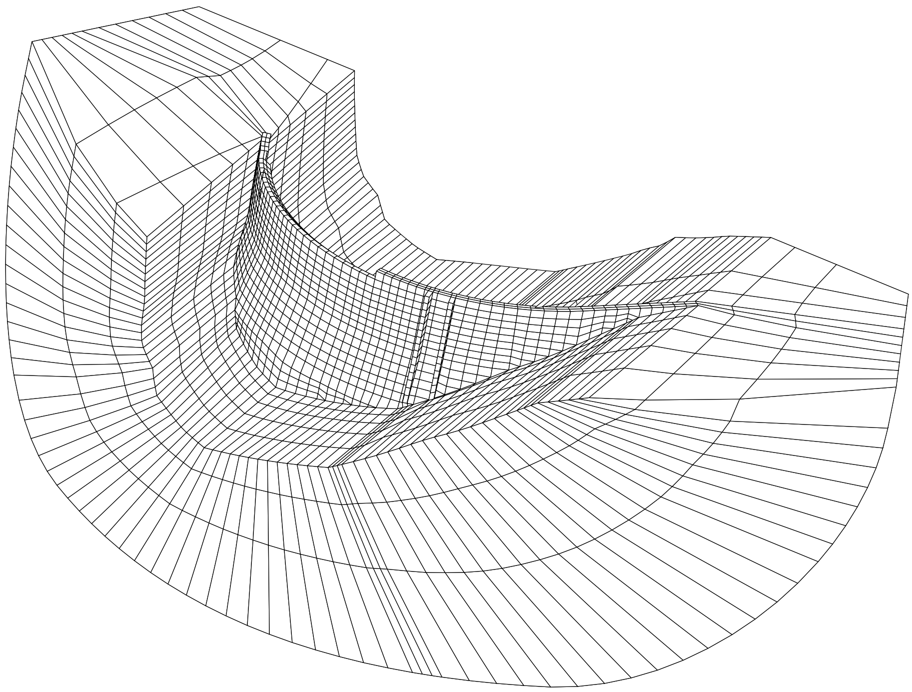


Figure 11 - Initial mesh of the dam, upstream side

The strategy of the adaptation is based on a succession of adaptation. The first calculation is made until time #1, when the first level of water is reached. At this point, the calculation is stopped and the distribution of the stresses and the displacement is examined. On every single element, the variation of the displacement between the element and its neighbours is computed and stored. The elements where this value is higher than the mean over the domain plus four times the square deviation are selected ([9], [10]). These selected elements are split at the beginning of the algorithm and the propagation is done until a conformal mesh is obtained. At this point, this new mesh is used for a second calculation from the very beginning of the building of the dam to the time #2 of the filling up. We operate the same mechanism until time #8 (see Figure 12). Doing this, the mesh is adapted for every position of the upper level of the water.

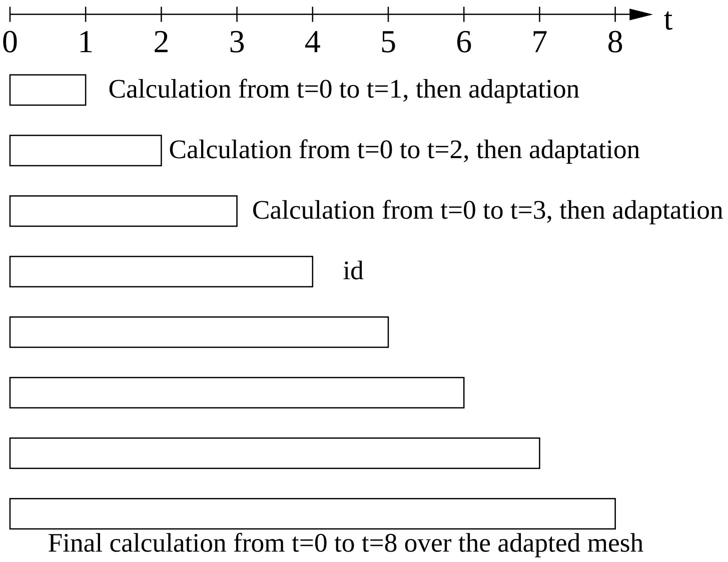


Figure 12 - The scheme of the adaptation

At the end of the process, the mesh is refined where it is necessary (see Figure 13), mainly at the centre and the bottom of the arch. The results of the simulation are similar to the reference. The number of degrees of freedom in the adapted mesh is much lower than in the reference mesh (see Table 1). This reduction allows using less memory and is very effective for large problems.

Mesh	Nodes	Hexahedra	Prisms	Tetrahedra	Pyramids
0	21 982	4 019	108	4	0
1	29 158	4 996	114	564	898
2	32 758	5 587	243	681	1 074
3	36 158	6 218	254	779	1 217
4	39 154	6 787	285	841	1 298
5	39 693	6 809	285	905	1 422
6	40 191	6 836	285	961	1 524
7	40 297	6 838	285	985	1 545
Reference	153 007	32 152	864	32	0

Table 1 - Number of elements during the adaptation

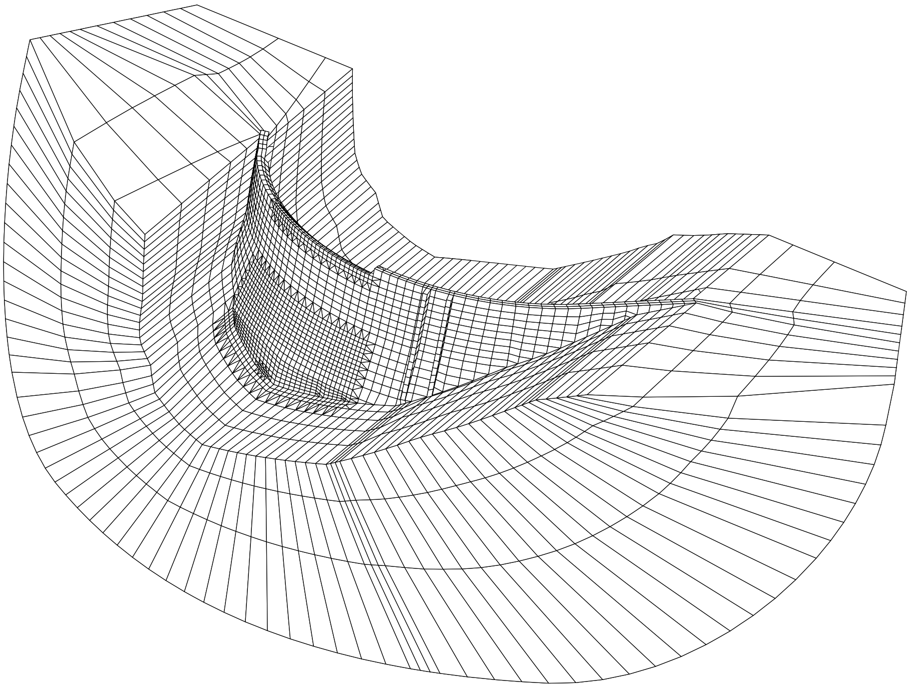


Figure 13 - Final mesh of the dam, upstream side

5. CONCLUSIONS

In this paper, we presented an h-refinement method for the adaptation of a conformal hexahedral mesh. The algorithm that propagates the decisions of splitting an element is governed by a simple rule on the faces: every quadrangular face that has at least two refined edges is to be refined. The conformal junction between two zones with a different level of refinement is achieved by a combination of tetrahedra and pyramids.

Some numerical experiments show that this technique is effective to increase the quality of the results. Thanks to this process, the confidence into the conclusion of a computation is improved.

REFERENCES

[1] S.H. Lo, D. Wu, K.Y. Sze, Adaptive meshing and analysis using transitional quadrilateral and hexahedral elements, *Finite Elements in Analysis and Design*, Vol. 46, pp. 2-16, 2010, doi:10.1016/j.finel.2009.06.010

[2] R. E. Bank, *PLTMG, A software package for solving elliptic partial differential equations, User's guide 6.0*, SIAM, Philadelphia, 1990.

[3] G. Nicolas and T. Fouquet, Adaptive Mesh Refinement for Conformal Hexahedral Meshes, *Finite Elements in Analysis and Design*, Vol. 67, pp. 1-12, 2013, doi:10.1016/j.finel.2012.11.008

[4] Salome web site: www.salome-platform.org

[5] G. Nicolas and T. Fouquet, Hexahedral mesh adaptation for finite volume methods, *ADMOS 07 Conference*, Goteborg 2007.
[6] *Code_Aster* web site: `www.code-aster.org`
[7] S. Prudhomme, J. T. Oden, On goal-oriented error estimation for elliptic problems: applications to the control of pointwise errors, *Comput. Methods Appl. Mech. Engrg.*, 176 (1999) 313-331.
[8] S. J. Owen, S. Saigal, Formation of pyramid elements for hexahedra to tetrahedral transitions, *Comput. Methods Appl. Mech. Engrg.*, 190 (2001) 4505-4518
[9] A. Chatziionnou, Y. Lu, A parallel adaptative finite element method for simulation of photon migration with the radiative-transfer-based model, *Numerical Methods in Engineering*, 25, Issue 6, pp. 751-770, 2009
[10] E. Shirani, M. Ameri, Combination of adaptative-grid-redistribustion and embedding methods, *Numerical Methods in Engineering*, 20, Issue 3, pp. 229-240, 2004