Advances in Engineering Software 102 (2016) 14–28

Contents lists available at ScienceDirect

Advances in Engineering Software

journal homepage: www.elsevier.com/locate/advengsoft

Research paper

# Improved adaptive mesh refinement for conformal hexahedral meshes

Gérald Nicolas[a,*], Thierry Fouquet[b], Samuel Geniaut[a], Sam Cuvilliez[a]

[a] EDF Lab Paris-Saclay, 7 boulevard Gaspard Monge, 91120 Palaiseau, France
[b] EDF – R&D, 6 quai Watier, 78400 Chatou, France

ARTICLE INFO

*Article history:*
Received 2 October 2015
Revised 19 July 2016
Accepted 19 July 2016

*Keywords:*
H-refinement
Adaptive meshing
Hexahedral mesh refinement
Conformal mesh
Mesh adaptation
Transition element
Pyramid

ABSTRACT

The h-refinement is a technique to achieve the mesh adaptation during a finite element simulation. Some elements are selected because the error is higher than a given threshold: they are split. This operation creates new elements where the size of the edges is divided by 2 in the zone of high error. To produce a conformal mesh, the connection between two zones with different levels of refinement must be examined. This paper proposes a solution in the case of meshes made of hexahedra. Depending on the type of interface, a set of patterns has been designed. In each pattern, the hexahedron is split into pyramids and tetrahedra. That specific splitting stops the propagation of the refinement and allows h-refinement in numerical simulations based on hexahedra with a conformal finite element method. At the end, an application to a crack analysis is shown.



## 1. Introduction

The numerical analysis is widely used to improve the efficiency of the industrial activities. These analyses are present in various stages of the processes: design, optimisation, non-destructive testing, simulation of accidents, etc. If a physical phenomenon is involved, the finite element method is often used to simulate the behaviour of an installation. Important decisions depend on the quality of the results of the simulations. Of course, many causes influence the quality of the results. One of these causes is the appropriateness between the mesh and the resolution. First, the mesh must be an accurate representation of the model. Then, on one hand, the number of elements must be as low as possible to reduce the cost of the simulation; on the other hand, the size of the elements must be small enough to increase the precision of the calculation. The mesh adaptation brings solutions to this problem.

The mesh adaptation is based on 2 stages if one wants to get the optimal size of the elements [1]. Firstly, a numerical analysis gives the indications of the error that is made on every element and it often suggests an optimal size of each element. For more than 3 decades, a large theoretical work has been made to compute such an error indicator [2–4]. Nevertheless, the definition and the implementation in an industrial software is still a hard task. This subject will not be discussed here. Secondly, a new mesh is created with these information; it is the so-called h-refinement. A solution consists in building a new mesh *ex nihilo*; but only a few mesh generators are able to automatically create a mesh that respects the constraints of size [5]. If the mesh has to be made of hexahedra, the problem challenges many researchers and an automatic procedure for industrial cases is not available yet [6–9], and many contributions from the International Meshing Roundtable [10]. Considering that difficulties, the h-refinement by division of the elements is a powerful solution [11–13]. The initial mesh is produced by a mesh generator. A calculation is made over this mesh and the error indicator is analysed to select the elements where the size should be smaller. Those elements are cut and a special treatment is applied to connect the refined zone and the unrefined zone to produce a new mesh that has to be conformal.

With a hexahedral mesh, the connection between two zones with different levels of refinement can be made either by special functions during the computation or by the introduction of tetrahedra and pyramids to avoid any modification of the solver resolution [14–18]. We developed a solution to build such a transition by a combination of tetrahedra and pyramids [19,20]. After using it in some industrial studies, we realized that the solution had a weak point. If the large errors were not localised through a direction parallel to the faces of the hexahedra, in other words if it were as a staircase, the refinement would spread too widely. Considering that drawback, the algorithm was improved and now the influence of the refinement is similar with a tetrahedral or a hexahedral

* Corresponding author.
*E-mail addresses:* gerald.nicolas@edf.fr (G. Nicolas), thierry.fouquet@edf.fr (T. Fouquet), samuel.geniaut@edf.fr (S. Geniaut), sam.cuvilliez@edf.fr (S. Cuvilliez).

http://dx.doi.org/10.1016/j.advengsoft.2016.07.014

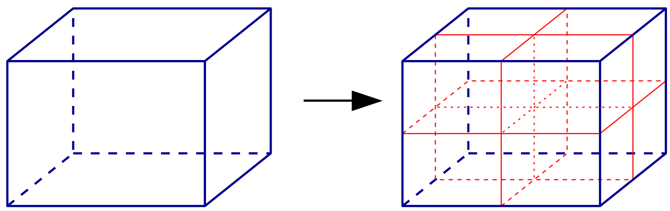

**Fig. 1.** Standard refinement of a hexahedron.

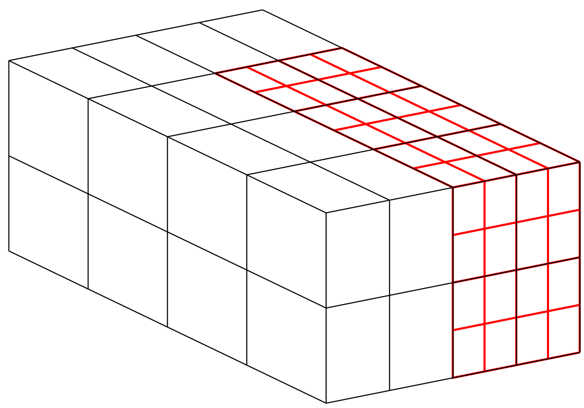

**Fig. 2.** A regularly refined zone inside a block.

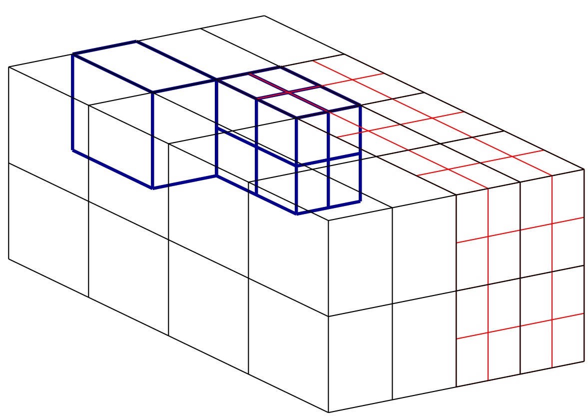

**Fig. 3.** Transition between 2 hexahedra.

**Table 1**
Procedure for the 3 or 4 pending nodes.

Define S as the stack of all the untagged faces.<br>
**while** (S is not empty) **do**<br>
    Examine F, the last face of S<br>
    **if** (the 4 edges of F are tagged) then<br>
        Tag F<br>
    **else if** (3 edges of F are tagged) then<br>
        Tag the 4th edge<br>
        Tag F<br>
        Add into S all the untagged faces that share this 4th edge.<br>
    **end if**<br>
    Remove the face F from the stack S<br>
**end**

mesh. This paper presents the full algorithm to produce conformal and adapted meshes with an h-refinement method. The exhaustive description of the transition meshes is given. Finally, an example shows the interest of the method in an industrial application.

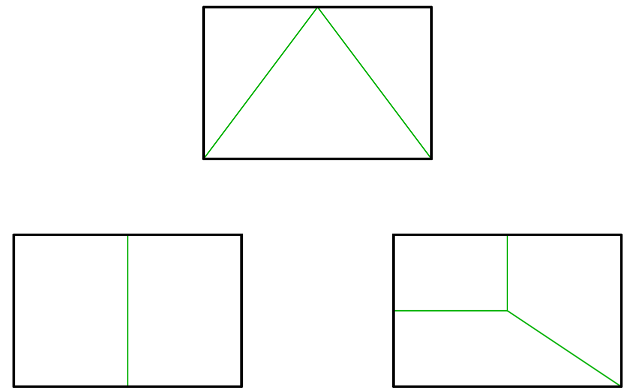

**Fig. 4.** Patterns for the refinement of a quadrangle.

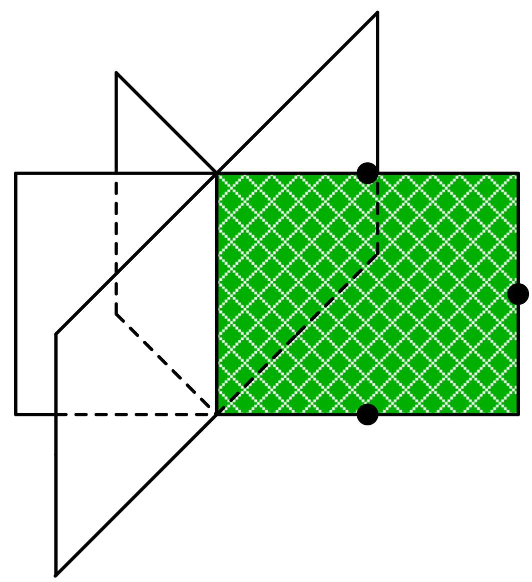

**Fig. 5.** A face with 3 tagged edges.

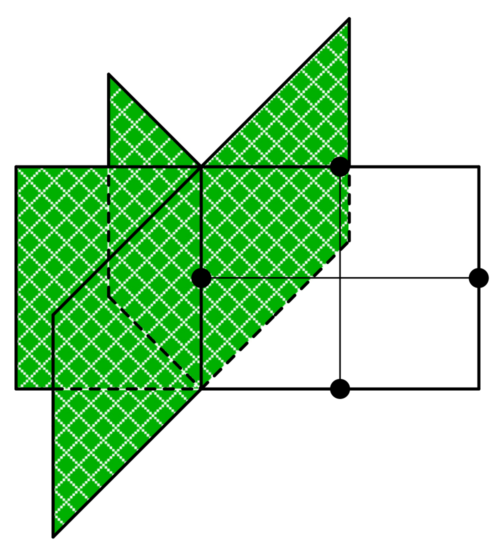

**Fig. 6.** Impact for the next faces.

**Table 2**
Number of cases vs number of split edges.

| Split edges | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|
| Cases | 1 | 4 | 8 | 13 | 9 | 7 | 2 | 2 | 1 |

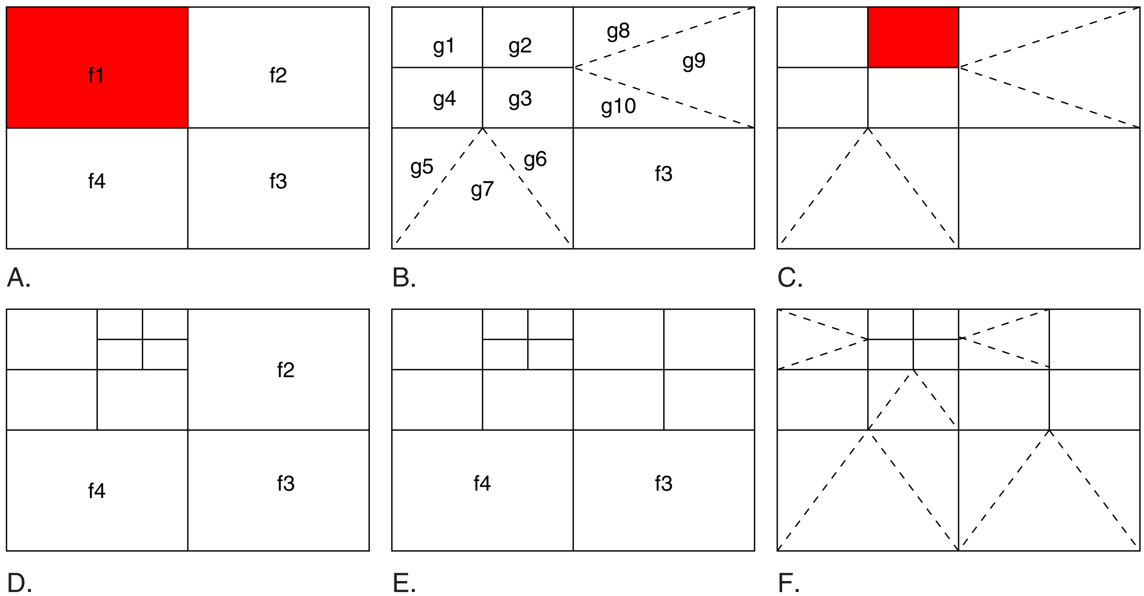


**Fig. 7.** Treatment of the conformity at level > 1.

**Table 3**
Global algorithm.

```
Transfer the error indicator onto the faces and the edges.
Remove all the transition elements
for all the levels from the highest to #0 do
        for all the faces and the edges in the current level do
                Propagate the refinement to get rid of 3 pending nodes in a face
                Propagate the refinement to the lower level if an edge is twice split
        end for
end for
Apply: standard refinement
Apply: transition refinement
```

## 2. The algorithm

### *2.1. Initialisation of the process*

At the end of a calculation over a hexahedral mesh, the solver computes an error indicator. This indicator can be an evaluation of the error, in the sense of a numerical analysis, and sometimes it provides information about the optimal size. But if this analysis is not available, the indicator may be a field over the mesh that represents any other data, such as the gradient of the solution, the distance from an object … It is not as accurate as a real error indicator but the important point is that the indicator is able to identify the zones where the elements are too large.

This indicator is expressed over the hexahedral elements. A threshold for the refinement is given together to drive the refinement. The first phase consists in sorting the elements. For each element, if the value of the error indicator is greater than the threshold, its 6 faces and its 12 edges are tagged. In some configurations, the error indicator may be expressed over the nodes. The initialisation is slightly modified. Each edge of the elements is considered. If the values of the error indicator over the two vertices of the edge are greater than the threshold, the edge is tagged.

At the end of this phase, some hexahedral elements are ready to be regularly refined: their 6 faces and their 12 edges are tagged. By placing a node at its midsection, each edge is equally split. A new node is added at the centre of each face and the quadrangular face is split into 4 quadrangles. Lastly, a new node is added at the centroid of the hexahedron and 8 new hexahedra are created by connecting this central node to the external nodes, edges and faces (see Fig. 1).

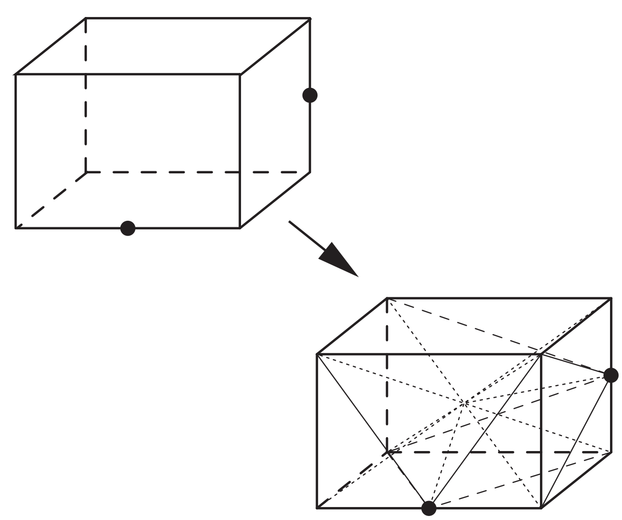

**Fig. 8.** A hexahedron with 2 cut edges.

Now, a fundamental problem appears in the transition between a refined hexahedron and an intact one. Imagine a block where a zone has been selected for the refinement by the indicator: all the hexahedra in that zone are going to be regularly refined (Fig. 2). But at the interface with the unrefined zone, one or more pending

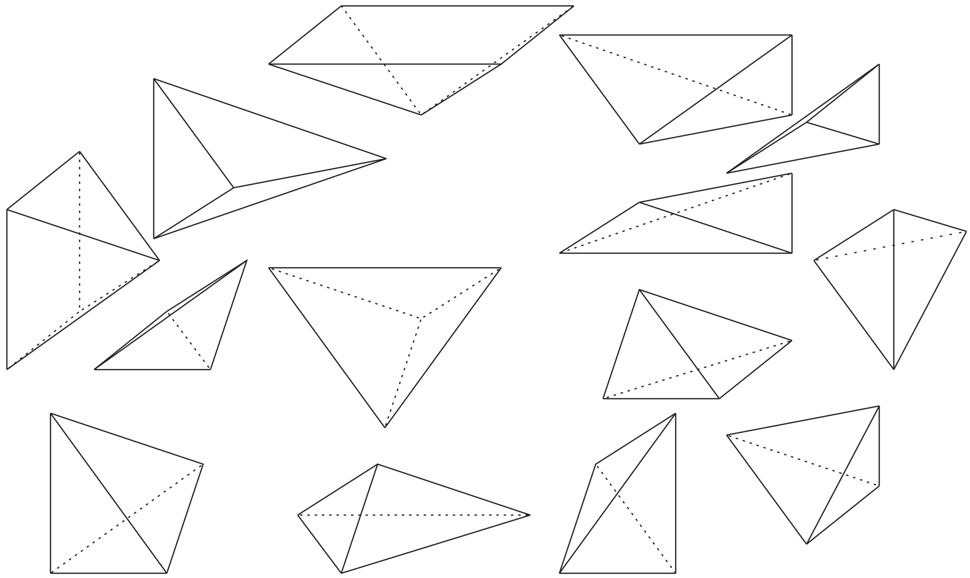

**Fig. 9.** Pyramids and tetrahedra - hexahedron with 2 cut edges.

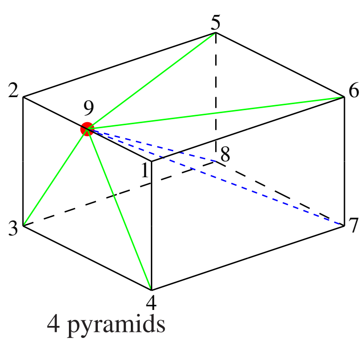


**Fig. 10.** With 1 edge subjected to division.

**Table 4**
With 1 edge subjected to division.

| | | | | | |
|---|---|---|---|---|---|
| P1 | 4 | 1 | 6 | 7 | 9 |
| P2 | 2 | 3 | 8 | 5 | 9 |
| P3 | 3 | 4 | 7 | 8 | 9 |
| P4 | 6 | 5 | 8 | 7 | 9 |

nodes are present (Fig. 3). The extreme solution would be to refine every hexahedron in such a situation. But this strategy would lead to a uniform refinement of the mesh, which is obviously not the objective of the mesh adaptation. A compromise consists in defining a special treatment for these zones with the examination of the status of every face.

### 2.2. The pending nodes on the quadrangular faces

In this phase of the algorithm, the hexahedra are forgotten and the treatment is concentrated on the quadrangular faces and their edges. Depending on the distribution of the error indicator, a face may have 1, 2, 3 or 4 pending nodes. We decided to use special templates for the cases with 1 or 2 pending nodes. Three patterns for the special refinements are required by dividing the quadrangular face into 3 triangles (1 pending node) or 2 quadrangles (2 opposite pending nodes) or 3 quadrangles (2 next pending nodes) (see Fig. 4). Some other solutions could be possible and this choice will be discussed in Section 4.1.

Considering this choice, we have to get rid of the cases of 3 or 4 pending nodes in a quadrangular face. A rule is defined: “tag each face where 3 or 4 edges are tagged” . All the untagged faces are considered. If their 4 edges are tagged, the face itself is tagged. If 3 edges are tagged, the face itself is tagged. The 4th edge is tagged as well. But the situation may have changed for the faces that share that 4th edge: they are examined to check their number of pending nodes. This treatment is applied until no face has

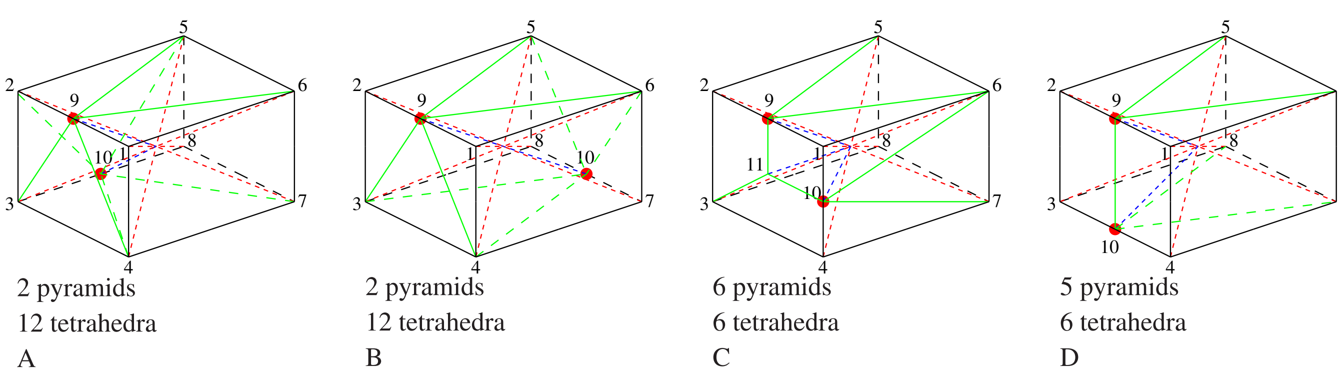


**Fig. 11.** With 2 edges subjected to division.

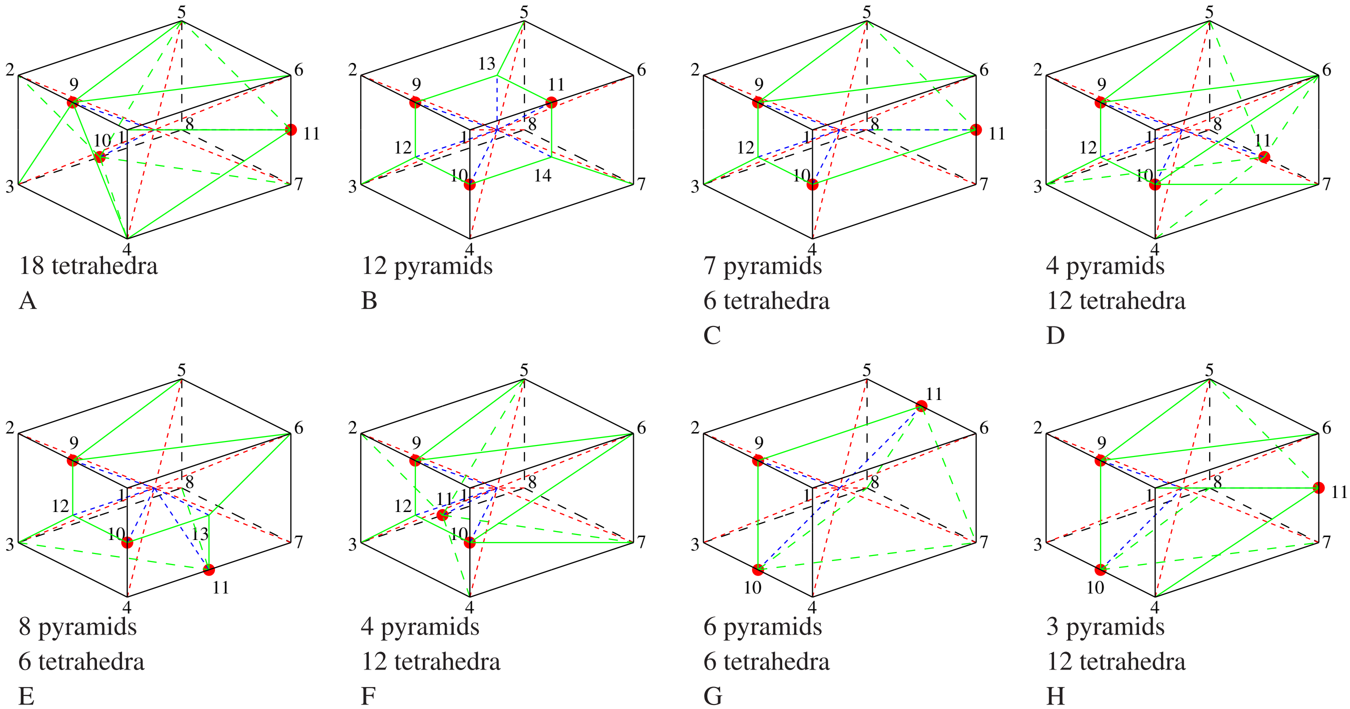


**Fig. 12.** With 3 edges subjected to division.

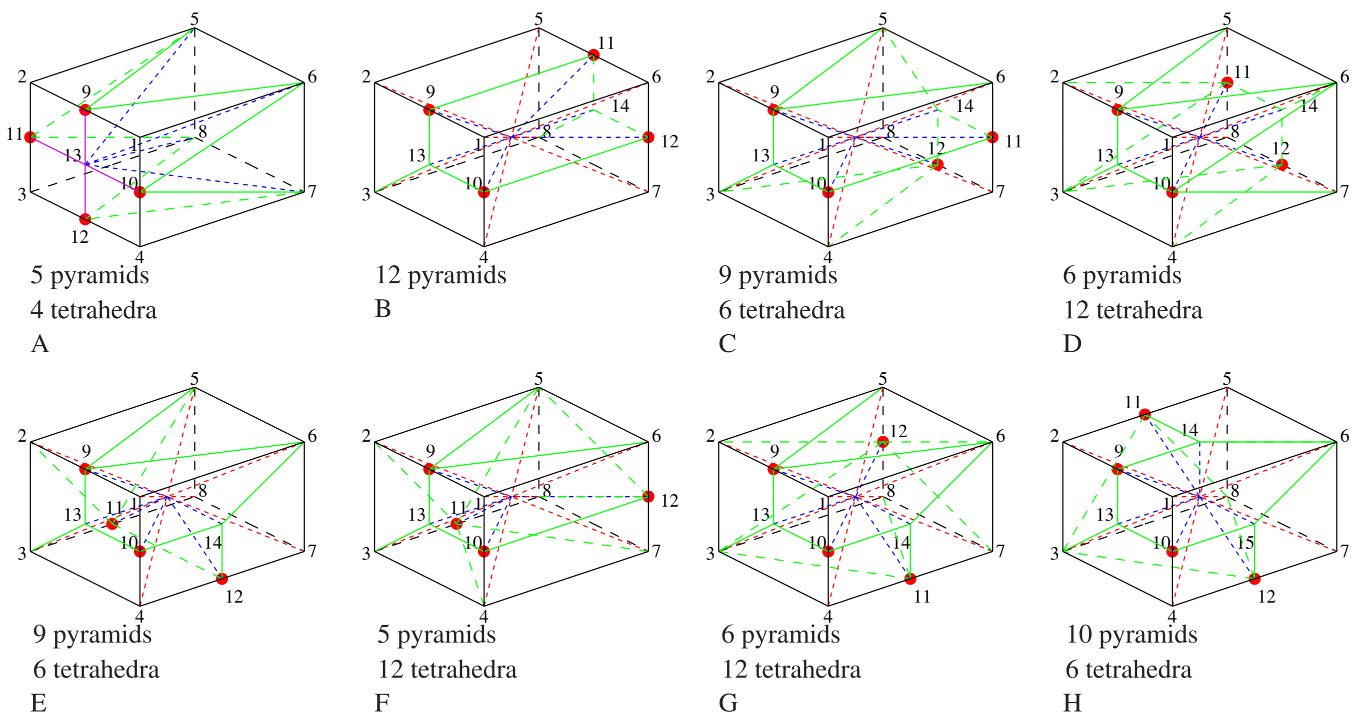


**Fig. 13.** With 4 edges subjected to division (1/2).

got neither 3 nor 4 pending nodes. Its application is described in Table 1.

Fig. 5 illustrates a case of a face that belongs to the stack of the untagged faces but with 3 tagged edges. After the treatment, the 4th edge and the face itself are tagged. The face is withdrawn from the stack. At the same time, all the next faces are included into the stack (Fig. 6).

### 2.3. Back to the hexahedra

At the end of this iterative procedure, each quadrangular face may be in one of the following status:

- No edge is tagged: the face is kept
- 1 edge is tagged: the face is split into 3 triangles
- 2 opposite edges are tagged: the face is split into 2 quadrangles
- 2 adjacent edges are tagged: the face is split into 3 quadrangles

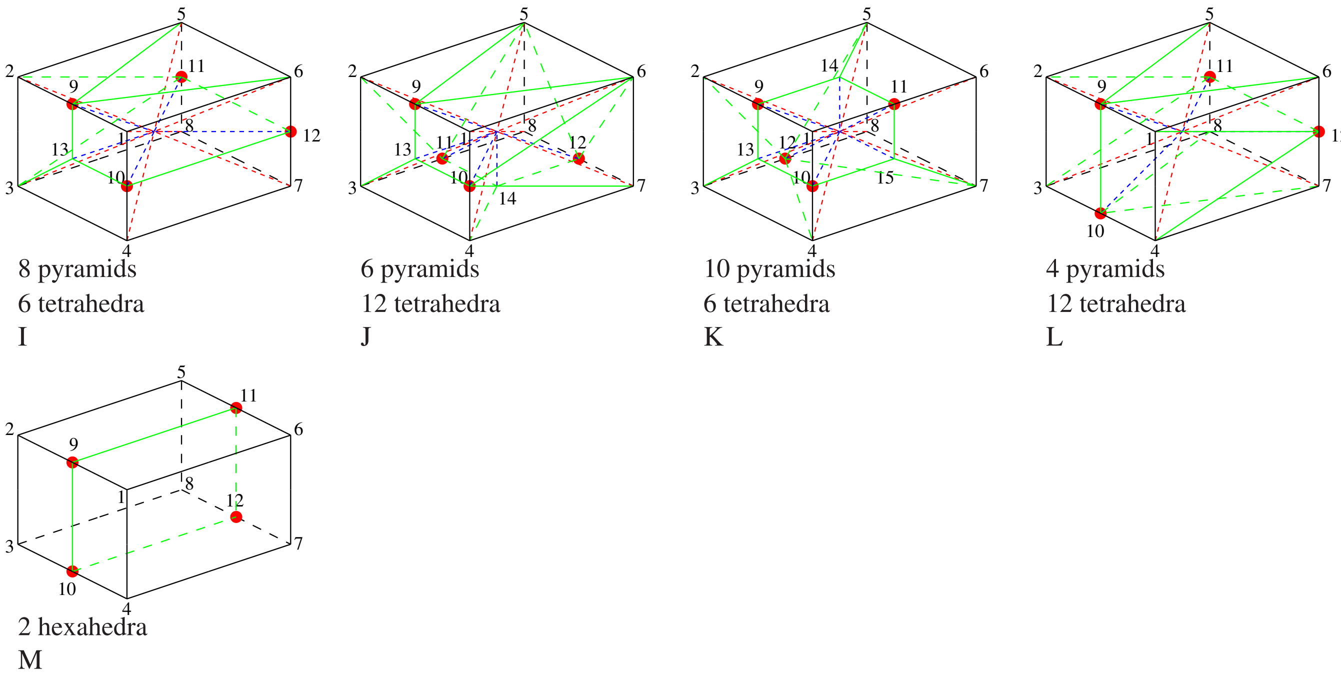


**Fig. 14.** With 4 edges subjected to division (2/2).

**Table 5**
With 2 edges subjected to division.

| A | | | | | | B | | | | | | C | | | | | | D | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T1 | 0 | 9 | 4 | 3 | | T1 | 0 | 9 | 4 | 3 | | T1 | 0 | 9 | 5 | 6 | | T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 1 | 4 | | T2 | 0 | 9 | 1 | 4 | | T2 | 0 | 9 | 2 | 5 | | T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 3 | 2 | | T3 | 0 | 9 | 3 | 2 | | T3 | 0 | 9 | 6 | 1 | | T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 9 | 5 | 6 | | T4 | 0 | 9 | 5 | 6 | | T4 | 0 | 10 | 6 | 7 | | T4 | 0 | 10 | 7 | 8 | |
| T5 | 0 | 9 | 2 | 5 | | T5 | 0 | 9 | 2 | 5 | | T5 | 0 | 10 | 1 | 6 | | T5 | 0 | 10 | 4 | 7 | |
| T6 | 0 | 9 | 6 | 1 | | T6 | 0 | 9 | 6 | 1 | | T6 | 0 | 10 | 7 | 4 | | T6 | 0 | 10 | 8 | 3 | |
| T7 | 0 | 10 | 5 | 2 | | T7 | 0 | 10 | 3 | 4 | | P1 | 1 | 10 | 11 | 9 | 0 | P1 | 1 | 4 | 10 | 9 | 0 |
| T8 | 0 | 10 | 8 | 5 | | T8 | 0 | 10 | 8 | 3 | | P2 | 4 | 3 | 11 | 10 | 0 | P2 | 3 | 2 | 9 | 10 | 0 |
| T9 | 0 | 10 | 2 | 3 | | T9 | 0 | 10 | 4 | 7 | | P3 | 2 | 9 | 11 | 3 | 0 | P3 | 4 | 1 | 6 | 7 | 0 |
| T10 | 0 | 10 | 4 | 7 | | T10 | 0 | 10 | 6 | 5 | | P4 | 2 | 3 | 8 | 5 | 0 | P4 | 2 | 3 | 8 | 5 | 0 |
| T11 | 0 | 10 | 3 | 4 | | T11 | 0 | 10 | 7 | 6 | | P5 | 3 | 4 | 7 | 8 | 0 | P5 | 6 | 5 | 8 | 7 | 0 |
| T12 | 0 | 10 | 7 | 8 | | T12 | 0 | 10 | 5 | 8 | | P6 | 6 | 5 | 8 | 7 | 0 | | | | | | |
| P1 | 4 | 1 | 6 | 7 | 0 | P1 | 4 | 1 | 6 | 7 | 0 | | | | | | | | | | | | |
| P2 | 6 | 5 | 8 | 7 | 0 | P2 | 2 | 3 | 8 | 5 | 0 | | | | | | | | | | | | |

**Table 6**
With 3 edges subjected to division (1/2).

| A | | | | | B | | | | | | C | | | | | | D | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T1 | 0 | 9 | 4 | 3 | P1 | 1 | 10 | 12 | 9 | 0 | T1 | 0 | 9 | 5 | 6 | | T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 1 | 4 | P2 | 4 | 3 | 12 | 10 | 0 | T2 | 0 | 9 | 2 | 5 | | T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 3 | 2 | P3 | 2 | 9 | 12 | 3 | 0 | T3 | 0 | 9 | 6 | 1 | | T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 9 | 5 | 6 | P4 | 1 | 9 | 13 | 11 | 0 | T4 | 0 | 11 | 5 | 8 | | T4 | 0 | 10 | 6 | 7 | |
| T5 | 0 | 9 | 2 | 5 | P5 | 2 | 5 | 13 | 9 | 0 | T5 | 0 | 11 | 6 | 5 | | T5 | 0 | 10 | 1 | 6 | |
| T6 | 0 | 9 | 6 | 1 | P6 | 6 | 11 | 13 | 5 | 0 | T6 | 0 | 11 | 8 | 7 | | T6 | 0 | 10 | 7 | 4 | |
| T7 | 0 | 11 | 4 | 1 | P7 | 1 | 11 | 14 | 10 | 0 | P1 | 1 | 10 | 12 | 9 | 0 | T7 | 0 | 11 | 3 | 4 | |
| T8 | 0 | 11 | 7 | 4 | P8 | 6 | 7 | 14 | 11 | 0 | P2 | 4 | 3 | 12 | 10 | 0 | T8 | 0 | 11 | 8 | 3 | |
| T9 | 0 | 11 | 1 | 6 | P9 | 4 | 10 | 14 | 7 | 0 | P3 | 2 | 9 | 12 | 3 | 0 | T9 | 0 | 11 | 4 | 7 | |
| T10 | 0 | 10 | 5 | 2 | P10 | 2 | 3 | 8 | 5 | 0 | P4 | 1 | 6 | 11 | 10 | 0 | T10 | 0 | 11 | 6 | 5 | |
| T11 | 0 | 10 | 8 | 5 | P11 | 3 | 4 | 7 | 8 | 0 | P5 | 7 | 4 | 10 | 11 | 0 | T11 | 0 | 11 | 7 | 6 | |
| T12 | 0 | 10 | 2 | 3 | P12 | 6 | 5 | 8 | 7 | 0 | P6 | 2 | 3 | 8 | 5 | 0 | T12 | 0 | 11 | 5 | 8 | |
| T13 | 0 | 10 | 4 | 7 | | | | | | | P7 | 3 | 4 | 7 | 8 | 0 | P1 | 1 | 10 | 12 | 9 | 0 |
| T14 | 0 | 10 | 3 | 4 | | | | | | | | | | | | | P2 | 4 | 3 | 12 | 10 | 0 |
| T15 | 0 | 10 | 7 | 8 | | | | | | | | | | | | | P3 | 2 | 9 | 12 | 3 | 0 |
| T16 | 0 | 11 | 5 | 8 | | | | | | | | | | | | | P4 | 2 | 3 | 8 | 5 | 0 |
| T17 | 0 | 11 | 6 | 5 | | | | | | | | | | | | | | | | | | |
| T18 | 0 | 11 | 8 | 7 | | | | | | | | | | | | | | | | | | |

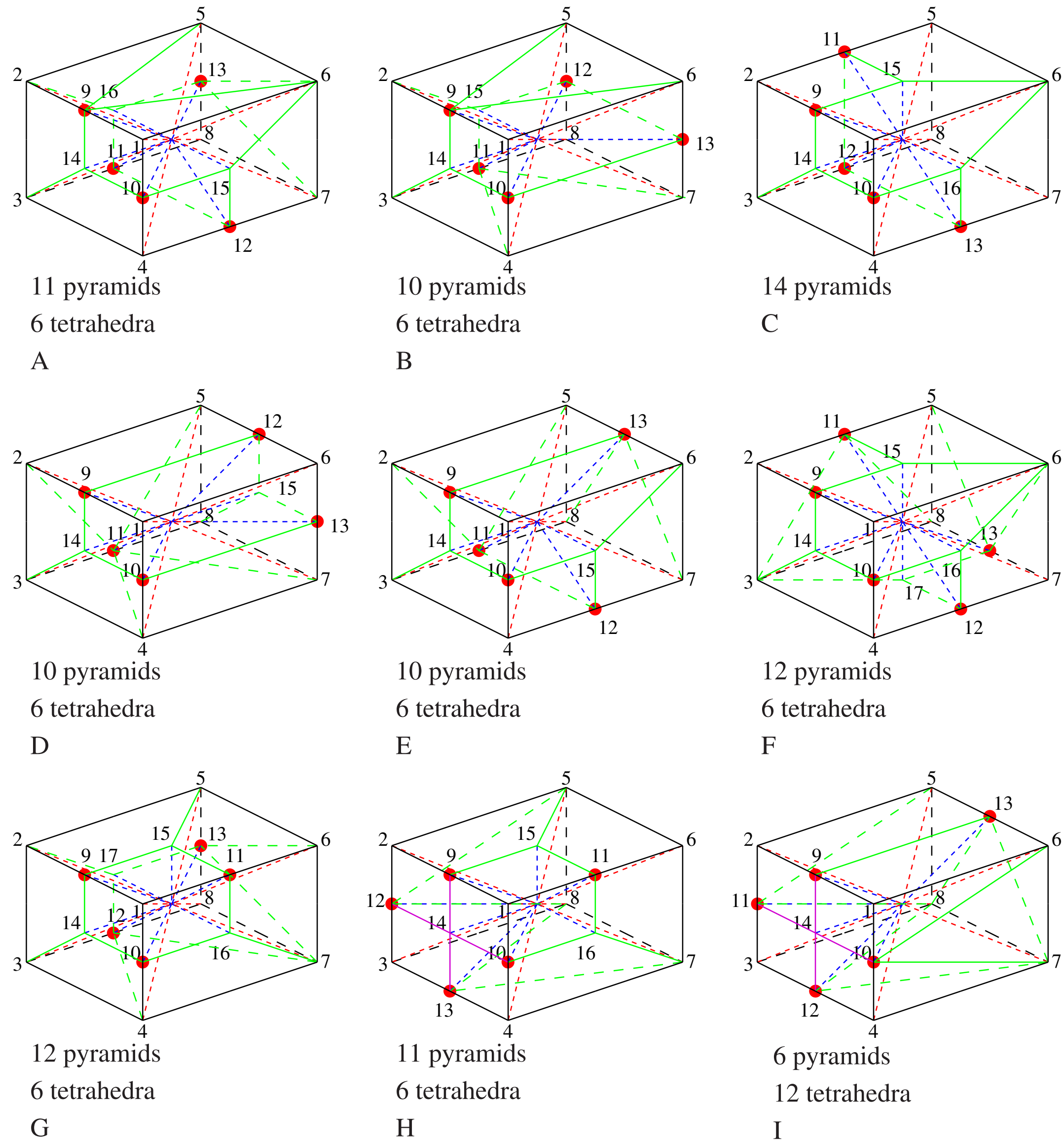


**Fig. 15.** With 5 edges subjected to division.

**Table 7**
With 3 edges subjected to division (2/2).

| E | | | | | | F | | | | | | G | | | | | | H | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | | T1 | 0 | 9 | 5 | 6 | | T1 | 0 | 10 | 7 | 8 | | T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | | T2 | 0 | 9 | 2 | 5 | | T2 | 0 | 10 | 4 | 7 | | T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | | T3 | 0 | 9 | 6 | 1 | | T3 | 0 | 10 | 8 | 3 | | T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 11 | 8 | 3 | | T4 | 0 | 10 | 6 | 7 | | T4 | 0 | 11 | 8 | 7 | | T4 | 0 | 11 | 4 | 1 | |
| T5 | 0 | 11 | 7 | 8 | | T5 | 0 | 10 | 1 | 6 | | T5 | 0 | 11 | 5 | 8 | | T5 | 0 | 11 | 7 | 4 | |
| T6 | 0 | 11 | 3 | 4 | | T6 | 0 | 10 | 7 | 4 | | T6 | 0 | 11 | 7 | 6 | | T6 | 0 | 11 | 1 | 6 | |
| P1 | 1 | 10 | 12 | 9 | 0 | T7 | 0 | 11 | 5 | 2 | | P1 | 1 | 4 | 10 | 9 | 0 | T7 | 0 | 10 | 7 | 8 | |
| P2 | 4 | 3 | 12 | 10 | 0 | T8 | 0 | 11 | 8 | 5 | | P2 | 3 | 2 | 9 | 10 | 0 | T8 | 0 | 10 | 4 | 7 | |
| P3 | 2 | 9 | 12 | 3 | 0 | T9 | 0 | 11 | 2 | 3 | | P3 | 2 | 5 | 11 | 9 | 0 | T9 | 0 | 10 | 8 | 3 | |
| P4 | 4 | 10 | 13 | 11 | 0 | T10 | 0 | 11 | 4 | 7 | | P4 | 6 | 1 | 9 | 11 | 0 | T10 | 0 | 11 | 5 | 8 | |
| P5 | 1 | 6 | 13 | 10 | 0 | T11 | 0 | 11 | 3 | 4 | | P5 | 4 | 1 | 6 | 7 | 0 | T11 | 0 | 11 | 6 | 5 | |
| P6 | 7 | 11 | 13 | 6 | 0 | T12 | 0 | 11 | 7 | 8 | | P6 | 2 | 3 | 8 | 5 | 0 | T12 | 0 | 11 | 8 | 7 | |
| P7 | 2 | 3 | 8 | 5 | 0 | P1 | 1 | 10 | 12 | 9 | 0 | | | | | | | P1 | 1 | 4 | 10 | 9 | 0 |
| P8 | 6 | 5 | 8 | 7 | 0 | P2 | 4 | 3 | 12 | 10 | 0 | | | | | | | P2 | 3 | 2 | 9 | 10 | 0 |
| | | | | | | P3 | 2 | 9 | 12 | 3 | 0 | | | | | | | P3 | 2 | 3 | 8 | 5 | 0 |
| | | | | | | P4 | 6 | 5 | 8 | 7 | 0 | | | | | | | | | | | | |

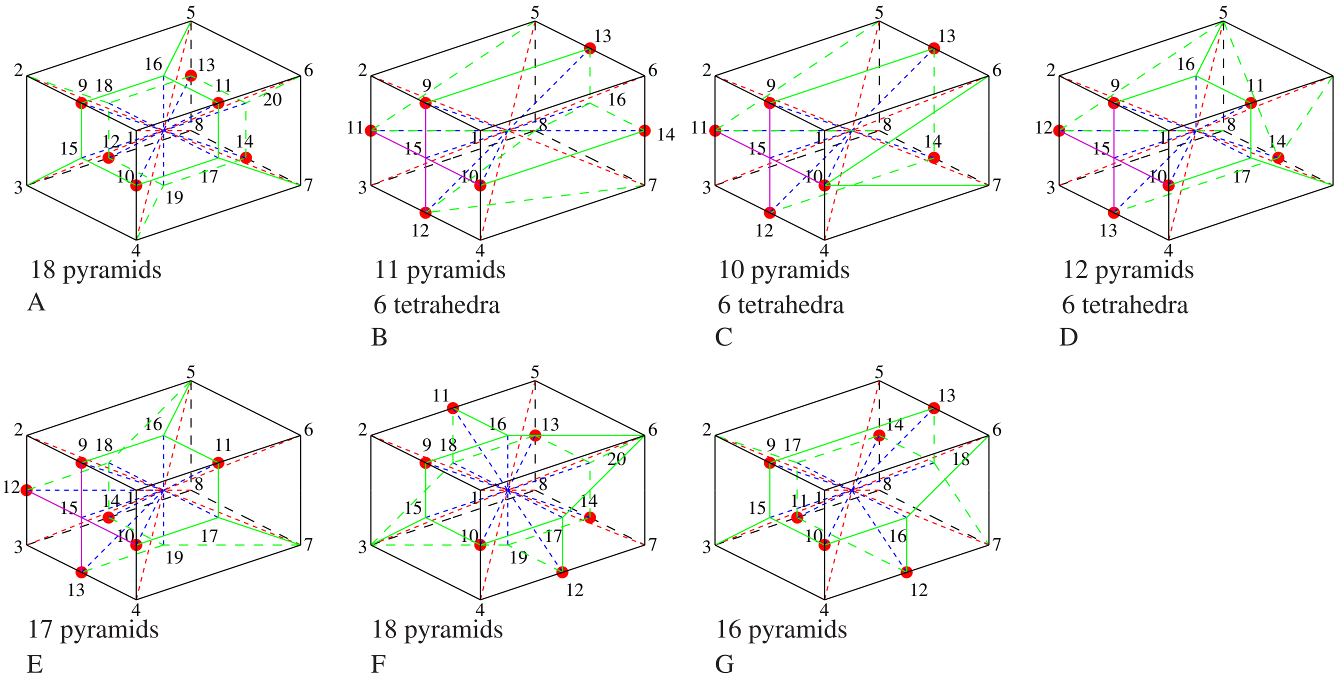


**Fig. 16.** With 6 edges subjected to division.

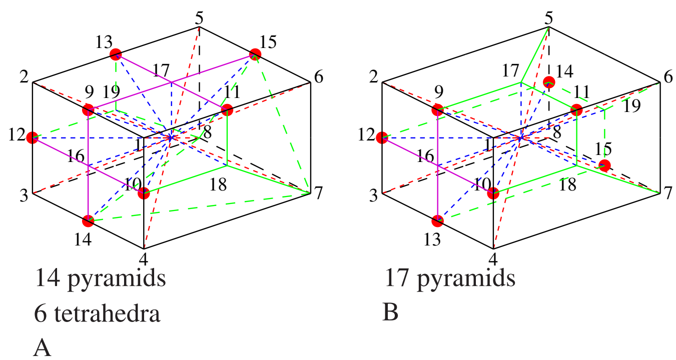


**Fig. 17.** With 7 edges subjected to division.

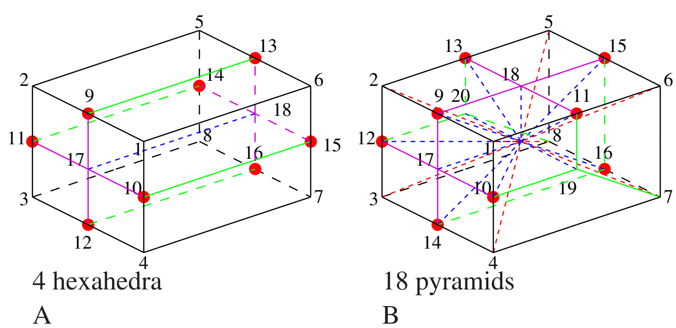


**Fig. 18.** With 8 edges subjected to division.

- 4 edges are tagged: the face is split into 4 quadrangles

Then, these patterns must be used to define the specific refinement of the hexahedra. When all the possible situations are examined for a given hexahedron, each of the 12 edges may be split or not. That means that *a priori* we have to deal with 2**12 = 4096 cases. These numerous cases are sorted firstly by the number of split edges, then by the pattern. First, we can remove the extremes

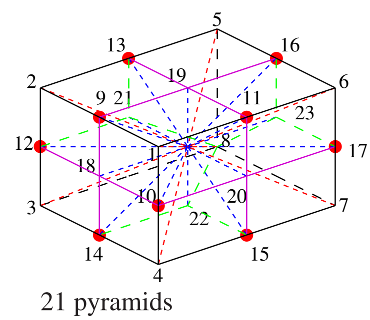


**Fig. 19.** With 9 edges subjected to division.

where 0 or 12 edges are split: that corresponds to an intact hexahedron or a regularly refined hexahedron. In the same way, 10 or 11 split edges cannot be found because it would violate the rule of splitting if 3 edges of a face are tagged. If 1 edge is split, 12 possibilities exist, depending on the split edge. We define a pattern with 1 split edge and the others 11 are deduced from this original pattern by rotations. If 2 edges are split, 4 possibilities exist: 2 edges on adjacent faces, 2 opposite edges, 2 adjacent edges on the same face, 2 opposite edges on the same face. Like in the case of a single split edge, each of these 4 categories contains similar situations that can be deduced one from the other by a set of rotations. The same analysis is made for 3 split edges, 4 split edges, etc. At the end, the consequence is a set of 47 different equivalence classes (see Table 2). All the corresponding patterns are described in the section #3.

*2.4. Next levels*

The algorithm was described for an initial mesh: the computation and the calculation of the error is made on the very first mesh #0, built by a mesh generator from CAD for example. With the adaptation procedure, a new mesh #1 is built with a mix of hexahedra, either from the initial mesh or from the standard splitting of the initial ones, pyramids and tetrahedra. The calculation

**Table 8**
With 4 edges subjected to division (1/3).

| A | | | | | |
|---|---|---|---|---|---|
| T1 | 13 | 9 | 5 | 6 | |
| T2 | 13 | 10 | 6 | 7 | |
| T3 | 13 | 12 | 7 | 8 | |
| T4 | 13 | 11 | 8 | 5 | |
| P1 | 6 | 5 | 8 | 7 | 13 |
| P2 | 1 | 10 | 13 | 9 | 6 |
| P3 | 4 | 12 | 13 | 10 | 7 |
| P4 | 3 | 11 | 13 | 12 | 8 |
| P5 | 2 | 9 | 13 | 11 | 5 |

| B | | | | | |
|---|---|---|---|---|---|
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 2 | 5 | 11 | 9 | 0 |
| P5 | 6 | 1 | 9 | 11 | 0 |
| P6 | 1 | 6 | 12 | 10 | 0 |
| P7 | 7 | 4 | 10 | 12 | 0 |
| P8 | 2 | 3 | 8 | 5 | 0 |
| P9 | 3 | 4 | 7 | 8 | 0 |
| P10 | 6 | 11 | 14 | 12 | 0 |
| P11 | 5 | 8 | 14 | 11 | 0 |
| P12 | 7 | 12 | 14 | 8 | 0 |

| C | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 12 | 3 | 4 | |
| T5 | 0 | 12 | 8 | 3 | |
| T6 | 0 | 12 | 4 | 7 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 1 | 6 | 11 | 10 | 0 |
| P5 | 7 | 4 | 10 | 11 | 0 |
| P6 | 2 | 3 | 8 | 5 | 0 |
| P7 | 7 | 11 | 14 | 12 | 0 |
| P8 | 6 | 5 | 14 | 11 | 0 |
| P9 | 8 | 12 | 14 | 5 | 0 |

| D | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 10 | 6 | 7 | |
| T5 | 0 | 10 | 1 | 6 | |
| T6 | 0 | 10 | 7 | 4 | |
| T7 | 0 | 11 | 2 | 3 | |
| T8 | 0 | 11 | 5 | 2 | |
| T9 | 0 | 11 | 3 | 8 | |
| T10 | 0 | 12 | 3 | 4 | |
| T11 | 0 | 12 | 8 | 3 | |
| T12 | 0 | 12 | 4 | 7 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 8 | 12 | 14 | 11 | 0 |
| P5 | 7 | 6 | 14 | 12 | 0 |
| P6 | 5 | 11 | 14 | 6 | 0 |

**Table 9**
With 4 edges subjected to division (2/3).

| E | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 11 | 5 | 2 | |
| T5 | 0 | 11 | 8 | 5 | |
| T6 | 0 | 11 | 2 | 3 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 4 | 10 | 14 | 12 | 0 |
| P5 | 1 | 6 | 14 | 10 | 0 |
| P6 | 7 | 12 | 14 | 6 | 0 |
| P7 | 7 | 8 | 11 | 12 | 0 |
| P8 | 3 | 4 | 12 | 11 | 0 |
| P9 | 6 | 5 | 8 | 7 | 0 |

| F | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 11 | 5 | 2 | |
| T5 | 0 | 11 | 8 | 5 | |
| T6 | 0 | 11 | 2 | 3 | |
| T7 | 0 | 11 | 4 | 7 | |
| T8 | 0 | 11 | 3 | 4 | |
| T9 | 0 | 11 | 7 | 8 | |
| T10 | 0 | 12 | 5 | 8 | |
| T11 | 0 | 12 | 6 | 5 | |
| T12 | 0 | 12 | 8 | 7 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 1 | 6 | 12 | 10 | 0 |
| P5 | 7 | 4 | 10 | 12 | 0 |

| G | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 12 | 2 | 3 | |
| T5 | 0 | 12 | 5 | 2 | |
| T6 | 0 | 12 | 3 | 8 | |
| T7 | 0 | 11 | 8 | 3 | |
| T8 | 0 | 11 | 7 | 8 | |
| T9 | 0 | 11 | 3 | 4 | |
| T10 | 0 | 12 | 7 | 6 | |
| T11 | 0 | 12 | 8 | 7 | |
| T12 | 0 | 12 | 6 | 5 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 4 | 10 | 14 | 11 | 0 |
| P5 | 1 | 6 | 14 | 10 | 0 |
| P6 | 7 | 11 | 14 | 6 | 0 |

| H | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 11 | 3 | 8 | |
| T2 | 0 | 11 | 2 | 3 | |
| T3 | 0 | 11 | 8 | 5 | |
| T4 | 0 | 12 | 8 | 3 | |
| T5 | 0 | 12 | 7 | 8 | |
| T6 | 0 | 12 | 3 | 4 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 2 | 11 | 14 | 9 | 0 |
| P5 | 5 | 6 | 14 | 11 | 0 |
| P6 | 1 | 9 | 14 | 6 | 0 |
| P7 | 4 | 10 | 15 | 12 | 0 |
| P8 | 1 | 6 | 15 | 10 | 0 |
| P9 | 7 | 12 | 15 | 6 | 0 |
| P10 | 6 | 5 | 8 | 7 | 0 |

**Table 10**
With 4 edges subjected to division (3/3).

| I | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 11 | 2 | 3 | |
| T5 | 0 | 11 | 5 | 2 | |
| T6 | 0 | 11 | 3 | 8 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 1 | 6 | 12 | 10 | 0 |
| P5 | 7 | 4 | 10 | 12 | 0 |
| P6 | 3 | 4 | 7 | 8 | 0 |
| P7 | 8 | 7 | 12 | 11 | 0 |
| P8 | 6 | 5 | 11 | 12 | 0 |

| J | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 10 | 6 | 7 | |
| T5 | 0 | 10 | 1 | 6 | |
| T6 | 0 | 10 | 7 | 4 | |
| T7 | 0 | 11 | 5 | 2 | |
| T8 | 0 | 11 | 8 | 5 | |
| T9 | 0 | 11 | 2 | 3 | |
| T10 | 0 | 12 | 6 | 5 | |
| T11 | 0 | 12 | 7 | 6 | |
| T12 | 0 | 12 | 5 | 8 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 8 | 11 | 14 | 12 | 0 |
| P5 | 3 | 4 | 14 | 11 | 0 |
| P6 | 7 | 12 | 14 | 4 | 0 |

| K | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 12 | 5 | 2 | |
| T2 | 0 | 12 | 8 | 5 | |
| T3 | 0 | 12 | 2 | 3 | |
| T4 | 0 | 12 | 4 | 7 | |
| T5 | 0 | 12 | 3 | 4 | |
| T6 | 0 | 12 | 7 | 8 | |
| P1 | 1 | 10 | 13 | 9 | 0 |
| P2 | 4 | 3 | 13 | 10 | 0 |
| P3 | 2 | 9 | 13 | 3 | 0 |
| P4 | 1 | 9 | 14 | 11 | 0 |
| P5 | 2 | 5 | 14 | 9 | 0 |
| P6 | 6 | 11 | 14 | 5 | 0 |
| P7 | 1 | 11 | 15 | 10 | 0 |
| P8 | 6 | 7 | 15 | 11 | 0 |
| P9 | 4 | 10 | 15 | 7 | 0 |
| P10 | 6 | 5 | 8 | 7 | 0 |

| L | | | | | |
|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | |
| T2 | 0 | 9 | 2 | 5 | |
| T3 | 0 | 9 | 6 | 1 | |
| T4 | 0 | 12 | 4 | 1 | |
| T5 | 0 | 12 | 7 | 4 | |
| T6 | 0 | 12 | 1 | 6 | |
| T7 | 0 | 11 | 2 | 3 | |
| T8 | 0 | 11 | 5 | 2 | |
| T9 | 0 | 11 | 3 | 8 | |
| T10 | 0 | 10 | 7 | 8 | |
| T11 | 0 | 10 | 4 | 7 | |
| T12 | 0 | 10 | 8 | 3 | |
| P1 | 1 | 4 | 10 | 9 | 0 |
| P2 | 3 | 2 | 9 | 10 | 0 |
| P3 | 8 | 7 | 12 | 11 | 0 |
| P4 | 6 | 5 | 11 | 12 | 0 |

| M | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| H1 | 1 | 9 | 10 | 4 | 11 | 6 | 7 | 12 |
| H2 | 9 | 2 | 3 | 10 | 5 | 11 | 12 | 8 |

**Table 11**
With 5 edges subjected to division (1/3).

| A | | | | | | B | | | | | | C | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T1 | 0 | 9 | 5 | 6 | | T1 | 0 | 9 | 5 | 6 | | P1 | 1 | 10 | 14 | 9 | 0 |
| T2 | 0 | 9 | 2 | 5 | | T2 | 0 | 9 | 2 | 5 | | P2 | 4 | 3 | 14 | 10 | 0 |
| T3 | 0 | 9 | 6 | 1 | | T3 | 0 | 9 | 6 | 1 | | P3 | 2 | 9 | 14 | 3 | 0 |
| T4 | 0 | 13 | 7 | 6 | | T4 | 0 | 11 | 4 | 7 | | P4 | 2 | 11 | 15 | 9 | 0 |
| T5 | 0 | 13 | 8 | 7 | | T5 | 0 | 11 | 3 | 4 | | P5 | 5 | 6 | 15 | 11 | 0 |
| T6 | 0 | 13 | 6 | 5 | | T6 | 0 | 11 | 7 | 8 | | P6 | 1 | 9 | 15 | 6 | 0 |
| P1 | 1 | 10 | 14 | 9 | 0 | P1 | 1 | 10 | 14 | 9 | 0 | P7 | 4 | 10 | 16 | 13 | 0 |
| P2 | 4 | 3 | 14 | 10 | 0 | P2 | 4 | 3 | 14 | 10 | 0 | P8 | 1 | 6 | 16 | 10 | 0 |
| P3 | 2 | 9 | 14 | 3 | 0 | P3 | 2 | 9 | 14 | 3 | 0 | P9 | 7 | 13 | 16 | 6 | 0 |
| P4 | 4 | 10 | 15 | 12 | 0 | P4 | 1 | 6 | 13 | 10 | 0 | P10 | 8 | 5 | 11 | 12 | 0 |
| P5 | 1 | 6 | 15 | 10 | 0 | P5 | 7 | 4 | 10 | 13 | 0 | P11 | 2 | 3 | 12 | 11 | 0 |
| P6 | 7 | 12 | 15 | 6 | 0 | P6 | 8 | 12 | 15 | 11 | 0 | P12 | 7 | 8 | 12 | 13 | 0 |
| P7 | 8 | 13 | 16 | 11 | 0 | P7 | 5 | 2 | 15 | 12 | 0 | P13 | 3 | 4 | 13 | 12 | 0 |
| P8 | 5 | 2 | 16 | 13 | 0 | P8 | 3 | 11 | 15 | 2 | 0 | P14 | 6 | 5 | 8 | 7 | 0 |
| P9 | 3 | 11 | 16 | 2 | 0 | P9 | 8 | 7 | 13 | 12 | 0 | | | | | | |
| P10 | 7 | 8 | 11 | 12 | 0 | P10 | 6 | 5 | 12 | 13 | 0 | | | | | | |
| P11 | 3 | 4 | 12 | 11 | 0 | | | | | | | | | | | | |

**Table 12**
With 5 edges subjected to division (2/3).

| D | | | | | | E | | | | | | F | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T1 | 0 | 11 | 5 | 2 | | T1 | 0 | 11 | 5 | 2 | | T1 | 0 | 11 | 3 | 8 | |
| T2 | 0 | 11 | 8 | 5 | | T2 | 0 | 11 | 8 | 5 | | T2 | 0 | 11 | 2 | 3 | |
| T3 | 0 | 11 | 2 | 3 | | T3 | 0 | 11 | 2 | 3 | | T3 | 0 | 11 | 8 | 5 | |
| T4 | 0 | 11 | 4 | 7 | | T4 | 0 | 13 | 8 | 7 | | T4 | 0 | 13 | 6 | 5 | |
| T5 | 0 | 11 | 3 | 4 | | T5 | 0 | 13 | 5 | 8 | | T5 | 0 | 13 | 7 | 6 | |
| T6 | 0 | 11 | 7 | 8 | | T6 | 0 | 13 | 7 | 6 | | T6 | 0 | 13 | 5 | 8 | |
| P1 | 1 | 10 | 14 | 9 | 0 | P1 | 1 | 10 | 14 | 9 | 0 | P1 | 1 | 10 | 14 | 9 | 0 |
| P2 | 4 | 3 | 14 | 10 | 0 | P2 | 4 | 3 | 14 | 10 | 0 | P2 | 4 | 3 | 14 | 10 | 0 |
| P3 | 2 | 9 | 14 | 3 | 0 | P3 | 2 | 9 | 14 | 3 | 0 | P3 | 2 | 9 | 14 | 3 | 0 |
| P4 | 2 | 5 | 12 | 9 | 0 | P4 | 2 | 5 | 13 | 9 | 0 | P4 | 2 | 11 | 15 | 9 | 0 |
| P5 | 6 | 1 | 9 | 12 | 0 | P5 | 6 | 1 | 9 | 13 | 0 | P5 | 5 | 6 | 15 | 11 | 0 |
| P6 | 1 | 6 | 13 | 10 | 0 | P6 | 4 | 10 | 15 | 12 | 0 | P6 | 1 | 9 | 15 | 6 | 0 |
| P7 | 7 | 4 | 10 | 13 | 0 | P7 | 1 | 6 | 15 | 10 | 0 | P7 | 4 | 10 | 16 | 12 | 0 |
| P8 | 6 | 12 | 15 | 13 | 0 | P8 | 7 | 12 | 15 | 6 | 0 | P8 | 1 | 6 | 16 | 10 | 0 |
| P9 | 5 | 8 | 15 | 12 | 0 | P9 | 7 | 8 | 11 | 12 | 0 | P9 | 7 | 12 | 16 | 6 | 0 |
| P10 | 7 | 13 | 15 | 8 | 0 | P10 | 3 | 4 | 12 | 11 | 0 | P10 | 7 | 13 | 17 | 12 | 0 |
| | | | | | | | | | | | | P11 | 8 | 3 | 17 | 13 | 0 |
| | | | | | | | | | | | | P12 | 4 | 12 | 17 | 3 | 0 |

on this mesh #1 will give a new field of error indicator. When the mesh #1 is going to be refined, a special treatment is applied to the specific elements that were created to make the conformal transition. They are never refined; if they were, that would cause a degradation of the quality of the mesh because the angles would decrease at each iteration. The solution consists in unrefining them at the beginning of the treatment. Imagine a 2D case with 4 quadrangles; one is selected for refinement (f1 in Fig. 7/A). This face is refined and its two neighbours are split following the adequate pattern (Fig. 7/B). Imagine that a new calculation is made over this mesh #1 and imagine that the analysis of the error indicator requests the refinement of one of the children (face g2) (Fig. 7/C). The first operation consists in removing the transition elements (triangles g5–g10) and then in refining this selected quadrangle g2 of the upper level (Fig. 7/D). But in the adjacent face (f2), the edge between the refined zone and the non-refined zone is split twice. In this case, we apply the rule: “if an edge is refined twice, the face as to be refined” (f2 in Fig. 7/E). Lastly, the pending nodes are examined and the specific transition elements are created (Fig. 7/F).

### 2.5. General algorithm

A level is given to every face: the faces of the initial mesh belong to the level #0. The faces built by the standard refinement of a face of the level #0 belong to the level #1, etc. Due to the 2 rules about the propagation of the refinement in the neighbouring of the elements, the algorithm can be applied whatever the last level of refinement is. Each level is examined, from the highest to the initial level #0. In a level, if there are 3 pending nodes on a face, the 4th edge of the face is tagged. When the level is over, the refinement is transferred to the upper level where an edge is twice split. Then, the upper level is examined. At the end, the refinement is processed to produce a conformal mesh (See Table 3).

## 3. The specific splitting of the hexahedron

In each equivalence class, the splitting of the hexahedra is made with internal pyramids and/or tetrahedra. Most of the time a node is created at the centroid of the hexahedron; this node is connected to the triangular or the quadrangular faces created on the external faces of the hexahedron. For example, consider the case of a hexahedron where 2 edges are cut on 2 adjacent faces (Fig. 8). Following the previously defined pattern, the faces sharing the split edges are cut into 3 triangles. The last 2 faces of the hexahedron remain intact. A node is inserted at the centroid. 10 new edges are created from every node on the external faces of the hexahedron: the 8 initial vertices and the 2 new midpoints of the split edges. 22 new triangular faces are created: their first edge is one of the edges on the external faces (10 initial edges, 4 edges produced by the splitting of the 2 initial edges and 8 new edges on the split faces) and the other 2 edges are 2 of the new internal

**Table 13**
With 5 edges subjected to division (3/3).

| G | | | | | | H | | | | | | I | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| T1 | 0 | 12 | 4 | 7 | | T1 | 0 | 12 | 8 | 5 | | T1 | 0 | 10 | 6 | 7 | |
| T2 | 0 | 12 | 3 | 4 | | T2 | 0 | 12 | 3 | 8 | | T2 | 0 | 10 | 1 | 6 | |
| T3 | 0 | 12 | 7 | 8 | | T3 | 0 | 12 | 5 | 2 | | T3 | 0 | 10 | 7 | 4 | |
| T4 | 0 | 13 | 7 | 6 | | T4 | 0 | 13 | 7 | 8 | | T4 | 0 | 11 | 8 | 5 | |
| T5 | 0 | 13 | 8 | 7 | | T5 | 0 | 13 | 4 | 7 | | T5 | 0 | 11 | 3 | 8 | |
| T6 | 0 | 13 | 6 | 5 | | T6 | 0 | 13 | 8 | 3 | | T6 | 0 | 11 | 5 | 2 | |
| P1 | 1 | 10 | 14 | 9 | 0 | P1 | 1 | 10 | 14 | 9 | 0 | T7 | 0 | 12 | 7 | 8 | |
| P2 | 4 | 3 | 14 | 10 | 0 | P2 | 4 | 13 | 14 | 10 | 0 | T8 | 0 | 12 | 4 | 7 | |
| P3 | 2 | 9 | 14 | 3 | 0 | P3 | 3 | 12 | 14 | 13 | 0 | T9 | 0 | 12 | 8 | 3 | |
| P4 | 1 | 9 | 15 | 11 | 0 | P4 | 2 | 9 | 14 | 12 | 0 | T10 | 0 | 13 | 8 | 7 | |
| P5 | 2 | 5 | 15 | 9 | 0 | P5 | 1 | 9 | 15 | 11 | 0 | T11 | 0 | 13 | 5 | 8 | |
| P6 | 6 | 11 | 15 | 5 | 0 | P6 | 2 | 5 | 15 | 9 | 0 | T12 | 0 | 13 | 7 | 6 | |
| P7 | 1 | 11 | 16 | 10 | 0 | P7 | 6 | 11 | 15 | 5 | 0 | P1 | 1 | 10 | 14 | 9 | 0 |
| P8 | 6 | 7 | 16 | 11 | 0 | P8 | 1 | 11 | 16 | 10 | 0 | P2 | 4 | 12 | 14 | 10 | 0 |
| P9 | 4 | 10 | 16 | 7 | 0 | P9 | 6 | 7 | 16 | 11 | 0 | P3 | 3 | 11 | 14 | 12 | 0 |
| P10 | 8 | 13 | 17 | 12 | 0 | P10 | 4 | 10 | 16 | 7 | 0 | P4 | 2 | 9 | 14 | 11 | 0 |
| P11 | 5 | 2 | 17 | 13 | 0 | P11 | 6 | 5 | 8 | 7 | 0 | P5 | 2 | 5 | 13 | 9 | 0 |
| P12 | 3 | 12 | 17 | 2 | 0 | | | | | | | P6 | 6 | 1 | 9 | 13 | 0 |

**Table 14**
With 6 edges subjected to division (1/2).

| A | | | | | | B | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| P1 | 1 | 10 | 15 | 9 | 0 | T1 | 0 | 11 | 8 | 5 | |
| P2 | 4 | 3 | 15 | 10 | 0 | T2 | 0 | 11 | 3 | 8 | |
| P3 | 2 | 9 | 15 | 3 | 0 | T3 | 0 | 11 | 5 | 2 | |
| P4 | 1 | 9 | 16 | 11 | 0 | T4 | 0 | 12 | 7 | 8 | |
| P5 | 2 | 5 | 16 | 9 | 0 | T5 | 0 | 12 | 4 | 7 | |
| P6 | 6 | 11 | 16 | 5 | 0 | T6 | 0 | 12 | 8 | 3 | |
| P7 | 1 | 11 | 17 | 10 | 0 | P1 | 1 | 10 | 15 | 9 | 0 |
| P8 | 6 | 7 | 17 | 11 | 0 | P2 | 4 | 12 | 15 | 10 | 0 |
| P9 | 4 | 10 | 17 | 7 | 0 | P3 | 3 | 11 | 15 | 12 | 0 |
| P10 | 8 | 13 | 18 | 12 | 0 | P4 | 2 | 9 | 15 | 11 | 0 |
| P11 | 5 | 2 | 18 | 13 | 0 | P5 | 2 | 5 | 13 | 9 | 0 |
| P12 | 3 | 12 | 18 | 2 | 0 | P6 | 6 | 1 | 9 | 13 | 0 |
| P13 | 8 | 12 | 19 | 14 | 0 | P7 | 1 | 6 | 14 | 10 | 0 |
| P14 | 3 | 4 | 19 | 12 | 0 | P8 | 7 | 4 | 10 | 14 | 0 |
| P15 | 7 | 14 | 19 | 4 | 0 | P9 | 6 | 13 | 16 | 14 | 0 |
| P16 | 8 | 14 | 20 | 13 | 0 | P10 | 5 | 8 | 16 | 13 | 0 |
| P17 | 7 | 6 | 20 | 14 | 0 | P11 | 7 | 14 | 16 | 8 | 0 |
| P18 | 5 | 13 | 20 | 6 | 0 | | | | | | |

| C | | | | | | D | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| T1 | 0 | 10 | 6 | 7 | | T1 | 0 | 12 | 8 | 5 | |
| T2 | 0 | 10 | 1 | 6 | | T2 | 0 | 12 | 3 | 8 | |
| T3 | 0 | 10 | 7 | 4 | | T3 | 0 | 12 | 5 | 2 | |
| T4 | 0 | 11 | 8 | 5 | | T4 | 0 | 14 | 6 | 5 | |
| T5 | 0 | 11 | 3 | 8 | | T5 | 0 | 14 | 7 | 6 | |
| T6 | 0 | 11 | 5 | 2 | | T6 | 0 | 14 | 5 | 8 | |
| P1 | 1 | 10 | 15 | 9 | 0 | P1 | 1 | 10 | 15 | 9 | 0 |
| P2 | 4 | 12 | 15 | 10 | 0 | P2 | 4 | 13 | 15 | 10 | 0 |
| P3 | 3 | 11 | 15 | 12 | 0 | P3 | 3 | 12 | 15 | 13 | 0 |
| P4 | 2 | 9 | 15 | 11 | 0 | P4 | 2 | 9 | 15 | 12 | 0 |
| P5 | 2 | 5 | 13 | 9 | 0 | P5 | 1 | 9 | 16 | 11 | 0 |
| P6 | 6 | 1 | 9 | 13 | 0 | P6 | 2 | 5 | 16 | 9 | 0 |
| P7 | 4 | 7 | 14 | 12 | 0 | P7 | 6 | 11 | 16 | 5 | 0 |
| P8 | 8 | 3 | 12 | 14 | 0 | P8 | 1 | 11 | 17 | 10 | 0 |
| P9 | 5 | 8 | 14 | 13 | 0 | P9 | 6 | 7 | 17 | 11 | 0 |
| P10 | 7 | 6 | 13 | 14 | 0 | P10 | 4 | 10 | 17 | 7 | 0 |
| | | | | | | P11 | 4 | 7 | 14 | 13 | 0 |
| | | | | | | P12 | 8 | 3 | 13 | 14 | 0 |

edges. Lastly, with the central node as a common vertex, from the 2 intact faces, 2 pyramids are built and from the triangles of the split faces, 12 tetrahedra are created (Fig. 9).

The exhaustive description of all the patterns used for the conformal refinement of a hexahedron is given in figures from Figs. 10–19. By convention, the new node that is created at the centre of the hexahedron is #0. This number is not included into the figures to increase the readability. The connectivity of every conformal element is given in the tables from Tables 4–18. Each pattern is identified firstly by the number of split edges, secondly by a letter A, B, C … This letter plays the same role in the tables and in the figures. The first column of a table enumerates the transition element with ‘T’ for tetrahedron, ‘P’ for pyramid, ‘H’ for hexahedron. The number is the rank of the element. A line gives the list of the nodes that define the element. For example, ‘P3’ introduces the line of the connectivity of the 3rd pyramid.

## 4. Comments on the choices for the algorithm

### *4.1. The patterns for the quadrangular faces*

The choice for the patterns for the transition refinement of the quadrangular faces was guided by this rule: as simple as it could be. We must keep in mind that the objective is the refinement of the hexahedron. The more complicated the refinement of the face is, the more complicated the refinement of the hexahedron is. Considering that point, we decided not to define a special refinement of a quadrangle with 3 pending nodes. It could have been efficient in a pure 2D mesh to stop the propagation of the refinement but it would have been too heavy in 3D. It is the same for the case with 1 pending node: we did not consider the template that introduces a node in the centre to make 2 quadrangles and 1 triangle because it would have increased the total number of nodes. Basically, we think that the shape of the conformal transition does not really matter: the differences between the 2 meshes will not produce any significant differences for the final results of the calculation.

On the other hand, it is possible to simplify the variety of patterns and to only keep the faces with 1 pending node. In such a solution, the faces with 2 pending nodes are treated like the faces with 3 pending nodes: all the edges are split and the face itself is split. The solution has a great advantage: instead of 47 different cases for the refinement of the hexahedron, only 5 cases remain. That was the first method we developed. But there is a drawback: if the elements where the error indicator is higher than the threshold are placed like a staircase, the refinement propagates until a convex envelope of elements is refined (See Fig. 20). In some of our applications, that phenomenon leads to a very heavy refine-

**Table 15**
With 6 edges subjected to division (2/2).

| E | | | | | | F | | | | | | G | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| P1 | 1 | 10 | 15 | 9 | 0 | P1 | 1 | 10 | 15 | 9 | 0 | P1 | 1 | 10 | 15 | 9 | 0 |
| P2 | 4 | 13 | 15 | 10 | 0 | P2 | 4 | 3 | 15 | 10 | 0 | P2 | 4 | 3 | 15 | 10 | 0 |
| P3 | 3 | 12 | 15 | 13 | 0 | P3 | 2 | 9 | 15 | 3 | 0 | P3 | 2 | 9 | 15 | 3 | 0 |
| P4 | 2 | 9 | 15 | 12 | 0 | P4 | 2 | 11 | 16 | 9 | 0 | P4 | 2 | 5 | 13 | 9 | 0 |
| P5 | 1 | 9 | 16 | 11 | 0 | P5 | 5 | 6 | 16 | 11 | 0 | P5 | 6 | 1 | 9 | 13 | 0 |
| P6 | 2 | 5 | 16 | 9 | 0 | P6 | 1 | 9 | 16 | 6 | 0 | P6 | 4 | 10 | 16 | 12 | 0 |
| P7 | 6 | 11 | 16 | 5 | 0 | P7 | 4 | 10 | 17 | 12 | 0 | P7 | 1 | 6 | 16 | 10 | 0 |
| P8 | 1 | 11 | 17 | 10 | 0 | P8 | 1 | 6 | 17 | 10 | 0 | P8 | 7 | 12 | 16 | 6 | 0 |
| P9 | 6 | 7 | 17 | 11 | 0 | P9 | 7 | 12 | 17 | 6 | 0 | P9 | 8 | 14 | 17 | 11 | 0 |
| P10 | 4 | 10 | 17 | 7 | 0 | P10 | 5 | 11 | 18 | 13 | 0 | P10 | 5 | 2 | 17 | 14 | 0 |
| P11 | 3 | 14 | 18 | 12 | 0 | P11 | 2 | 3 | 18 | 11 | 0 | P11 | 3 | 11 | 17 | 2 | 0 |
| P12 | 8 | 5 | 18 | 14 | 0 | P12 | 8 | 13 | 18 | 3 | 0 | P12 | 7 | 8 | 11 | 12 | 0 |
| P13 | 2 | 12 | 18 | 5 | 0 | P13 | 7 | 14 | 19 | 12 | 0 | P13 | 3 | 4 | 12 | 11 | 0 |
| P14 | 3 | 13 | 19 | 14 | 0 | P14 | 8 | 3 | 19 | 14 | 0 | P14 | 5 | 14 | 18 | 13 | 0 |
| P15 | 4 | 7 | 19 | 13 | 0 | P15 | 4 | 12 | 19 | 3 | 0 | P15 | 8 | 7 | 18 | 14 | 0 |
| P16 | 8 | 14 | 19 | 7 | 0 | P16 | 8 | 14 | 20 | 13 | 0 | P16 | 6 | 13 | 18 | 7 | 0 |
| P17 | 6 | 5 | 8 | 7 | 0 | P17 | 7 | 6 | 20 | 14 | 0 | | | | | | |
| | | | | | | P18 | 5 | 13 | 20 | 6 | 0 | | | | | | |

**Table 16**
With 7 edges subjected to division.

| A | | | | | | B | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| T1 | 0 | 14 | 7 | 8 | | P1 | 1 | 10 | 16 | 9 | 0 |
| T2 | 0 | 14 | 4 | 7 | | P2 | 4 | 13 | 16 | 10 | 0 |
| T3 | 0 | 14 | 8 | 3 | | P3 | 3 | 12 | 16 | 13 | 0 |
| T4 | 0 | 15 | 8 | 7 | | P4 | 2 | 9 | 16 | 12 | 0 |
| T5 | 0 | 15 | 5 | 8 | | P5 | 1 | 9 | 17 | 11 | 0 |
| T6 | 0 | 15 | 7 | 6 | | P6 | 2 | 5 | 17 | 9 | 0 |
| P1 | 1 | 10 | 16 | 9 | 0 | P7 | 6 | 11 | 17 | 5 | 0 |
| P2 | 4 | 14 | 16 | 10 | 0 | P8 | 1 | 11 | 18 | 10 | 0 |
| P3 | 3 | 12 | 16 | 14 | 0 | P9 | 6 | 7 | 18 | 11 | 0 |
| P4 | 2 | 9 | 16 | 12 | 0 | P10 | 4 | 10 | 18 | 7 | 0 |
| P5 | 2 | 13 | 17 | 9 | 0 | P11 | 3 | 8 | 14 | 12 | 0 |
| P6 | 5 | 15 | 17 | 13 | 0 | P12 | 5 | 2 | 12 | 14 | 0 |
| P7 | 6 | 11 | 17 | 15 | 0 | P13 | 4 | 7 | 15 | 13 | 0 |
| P8 | 1 | 9 | 17 | 11 | 0 | P14 | 8 | 3 | 13 | 15 | 0 |
| P9 | 1 | 11 | 18 | 10 | 0 | P15 | 8 | 15 | 19 | 14 | 0 |
| P10 | 6 | 7 | 18 | 11 | 0 | P16 | 7 | 6 | 19 | 15 | 0 |
| P11 | 4 | 10 | 18 | 7 | 0 | P17 | 5 | 14 | 19 | 6 | 0 |
| P12 | 2 | 12 | 19 | 13 | 0 | | | | | | |
| P13 | 3 | 8 | 19 | 12 | 0 | | | | | | |
| P14 | 5 | 13 | 19 | 8 | 0 | | | | | | |

**Table 18**
With 9 edges subjected to division.

| | | | | | |
|---|---|---|---|---|---|
| P1 | 1 | 10 | 18 | 9 | 0 |
| P2 | 4 | 14 | 18 | 10 | 0 |
| P3 | 3 | 12 | 18 | 14 | 0 |
| P4 | 2 | 9 | 18 | 12 | 0 |
| P5 | 2 | 13 | 19 | 9 | 0 |
| P6 | 5 | 16 | 19 | 13 | 0 |
| P7 | 6 | 11 | 19 | 16 | 0 |
| P8 | 1 | 9 | 19 | 11 | 0 |
| P9 | 6 | 17 | 20 | 11 | 0 |
| P10 | 7 | 15 | 20 | 17 | 0 |
| P11 | 4 | 10 | 20 | 15 | 0 |
| P12 | 1 | 11 | 20 | 10 | 0 |
| P13 | 2 | 12 | 21 | 13 | 0 |
| P14 | 3 | 8 | 21 | 12 | 0 |
| P15 | 5 | 13 | 21 | 8 | 0 |
| P16 | 4 | 15 | 22 | 14 | 0 |
| P17 | 7 | 8 | 22 | 15 | 0 |
| P18 | 3 | 14 | 22 | 8 | 0 |
| P19 | 6 | 16 | 23 | 17 | 0 |
| P20 | 5 | 8 | 23 | 16 | 0 |
| P21 | 7 | 17 | 23 | 8 | 0 |

**Table 17**
With 8 edges subjected to division.

| A | | | | | | | | | B | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| H1 | 1 | 9 | 17 | 10 | 13 | 6 | 15 | 18 | P1 | 1 | 10 | 17 | 9 | 0 |
| H2 | 10 | 17 | 12 | 4 | 18 | 15 | 7 | 16 | P2 | 4 | 14 | 17 | 10 | 0 |
| H3 | 17 | 11 | 3 | 12 | 14 | 18 | 16 | 8 | P3 | 3 | 12 | 17 | 14 | 0 |
| H4 | 9 | 2 | 11 | 17 | 5 | 14 | 18 | 14 | P4 | 2 | 9 | 17 | 12 | 0 |
| | | | | | | | | | P5 | 2 | 13 | 18 | 9 | 0 |
| | | | | | | | | | P6 | 5 | 15 | 18 | 13 | 0 |
| | | | | | | | | | P7 | 6 | 11 | 18 | 15 | 0 |
| | | | | | | | | | P8 | 1 | 9 | 18 | 11 | 0 |
| | | | | | | | | | P9 | 1 | 11 | 19 | 10 | 0 |
| | | | | | | | | | P10 | 6 | 7 | 19 | 11 | 0 |
| | | | | | | | | | P11 | 4 | 10 | 19 | 7 | 0 |
| | | | | | | | | | P12 | 2 | 12 | 20 | 13 | 0 |
| | | | | | | | | | P13 | 3 | 8 | 20 | 12 | 0 |
| | | | | | | | | | P14 | 5 | 13 | 20 | 8 | 0 |
| | | | | | | | | | P15 | 4 | 7 | 16 | 14 | 0 |
| | | | | | | | | | P16 | 8 | 3 | 14 | 16 | 0 |
| | | | | | | | | | P17 | 5 | 8 | 16 | 15 | 0 |
| | | | | | | | | | P18 | 7 | 6 | 15 | 16 | 0 |

**Table 19**
Comparison of the efficiency.

| Strategy | Basic | Proposed |
|---|---|---|
| Number of nodes | 106,087 | 40,465 |
| Number of elements | 134,292 | 65,588 |
| CPU time (s) | 208 | 51 |

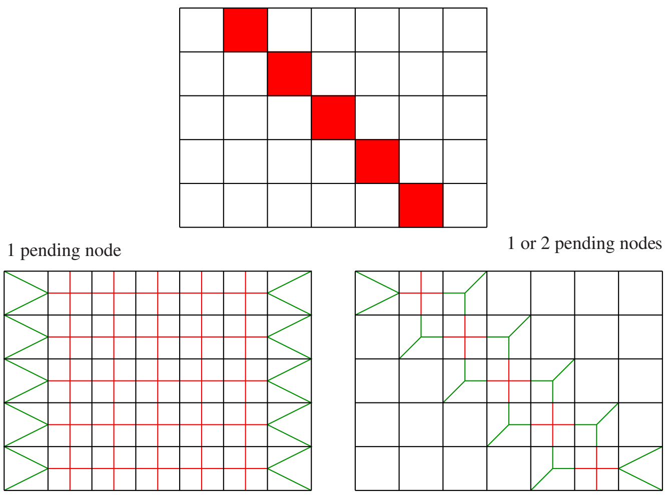


**Fig. 20.** Example of a staircase problem.

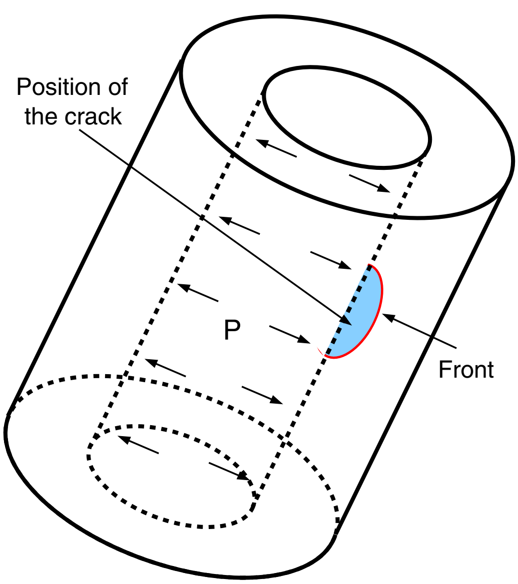


**Fig. 21.** Semi-elliptical crack in a pipe under internal pressure.

ment. The mesh adaptation was useless. Following our experience, the choice of 1 or 2 pending nodes is a better solution for efficiency and performance.

In some cases, the mesh adaptation is driven by an error estimator that provides the optimal size for every elements. For some of these elements, splitting the edge into 2 parts may be not the targeted size. Splitting into 3, 4 … parts would decrease the size of the elements but the impact on 3D and the implementation would be very hard. We decided to follow our rule “as simple as possible”, and we kept the solution of a 2 part splitting. More, if an error estimator provides information about the optimal size, we use it by applying several times the algorithm with an increasing threshold to catch the targeted elements.

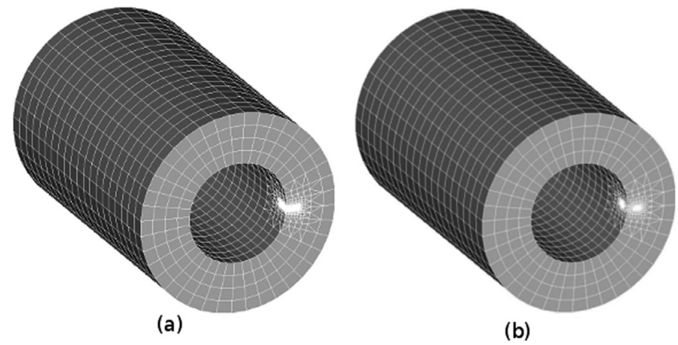


**Fig. 22.** Refined mesh of the pipe a) old b) new.

### 4.2. Implementation

The HOMARD software is designed to the mesh adaptation of every mesh we use in our finite element calculations. The mesh may be a pure 2D or 3D mesh or it may mix the dimensions. The type of elements can be mixed as well. A mesh made of both triangles and quadrangles may be treated. In 3D, the mesh can be made either of tetrahedra, pyramids or hexahedra or their combination. The solution to have the same routines comes from the algorithm itself. The initialisation consists in transferring the decisions from the error indicator to the faces and the edges. Then, the application of the rule to the neighbouring and the level is made through the faces, whatever they are, triangles or quadrangles. The only difference is the limit for the number of the pending nodes: 1 or 2 for a quadrangle as explained in this paper, 1 for a triangle. At the end, the 3D transition elements are created.

The method described in this paper deals with 47 patterns to refine a hexahedron and each pattern is used in many situations that are deduced by rotations. To prevent the risk of error in coding this complex part of the adaptation, we built a specific program in python. This program examines all the $2^{**}12 = 4096$ situations and it sorts them as discussed before. More, it writes the routines that effectively create the transition elements for each useful situation. At the end, these automatically written routines are linked to the main program HOMARD. Doing that, we guarantee that no situation has been forgotten and that we did not make error during the implementation.

## 5. Applications

The following application of the conformal refinement deals with fracture mechanics. Crack analyses with the Finite Element Method are of great interest in many industrial applications (nuclear, aeronautics…) at the design level or in-service. In the classical Griffith’s theory, the mechanical stress field is singular (infinite) on the crack front (the line delimiting the crack). To be able to precisely describe this high gradient, the mesh must be very well refined near the crack front. The method described in this paper is therefore well suited to refine the mesh near the crack front. We will represent the crack in the framework of the extended finite element method (X-FEM) [21,22]. In such a method, the crack is not explicitly meshed but it is represented by level set functions regardless of the mesh. The level set function is the signed distance to the crack. Then the mesh of the structure is refined in the zones near the crack: a pseudo-error indicator is built considering the distance to the crack front and a threshold is given to be coherent with the optimal size. The mesh adaptation as described in this paper refines the mesh until the optimal size is reached in a torus around the crack front; then the simulation of the mechani-

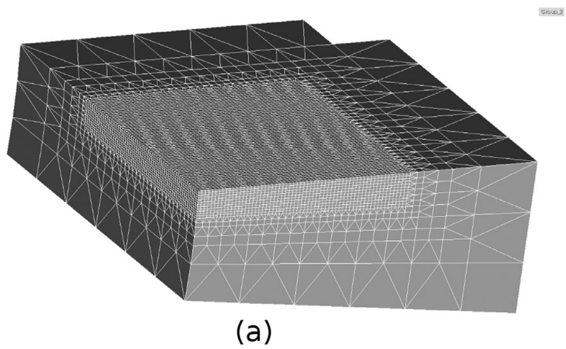


**Fig. 23.** Zoom around the crack front a) old b) new.

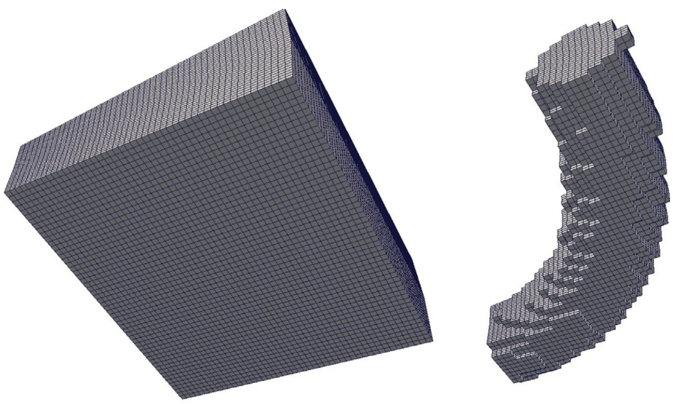

**Fig. 24.** Smallest elements a) old b) new.

cal response on the cracked structure is led to determine the stress and displacement fields.

We consider here a longitudinal semi-elliptical planar crack located in a pipe submitted to an internal pressure. The crack is perpendicular to the internal surface of the pipe (see Fig. 21). Due to symmetry consideration, only one half of the pipe (half-space $z \geq 0$) is considered. The initial mesh is made of hexahedra. This initial mesh is too coarse in the vicinity of the crack front, compared to its size, to get accurate results. A mesh refinement is then necessary. We will compare two mesh refinement strategies: (i) the first one is a strategy in which the transition between a refined hexahedron and an intact one is very simple (basic strategy) (ii) the second one is the strategy described in this paper. As it is said in the Section 4.1, the first strategy leads to a uniform refinement in a large area around the whole crack. On the contrary, the strategy described in this paper limits the refined area near the crack front. A comparison of the refined mesh and a zoom near the crack front obtained with the two strategies is presented in Figs. 22 and 23. The smallest elements are shown in Fig. 24: they are concentrated near the crack with the new technique. It can be clearly seen that for a given target element size, our strategy gives the optimal number of nodes. The computational cost is then strongly reduced. The comparison of the number of nodes, elements and computational time of the mechanical resolution is given in Table 19.

It is now important to be sure that the accuracy of the mechanical simulation with the optimal mesh is not deteriorated. To validate the refined mesh obtained with our strategy, a computation of the energy release rate along the crack front is done. The energy release rate ($G$) is a key parameter in fracture mechanics. The value of this quantity determines if the crack will propagate or not. The computation of $G$ is a post-processing of the stress and displacement fields that are obtained by the mechanical simulation. The computation of the stress and displacement fields and the post-processing of the energy release rate are done using *Code_Aster* [23]. We note $s$ the curvilinear coordinate along the crack front. The value of the computed energy release rate along the crack front is then compared between the two mesh refinement strategies (see Fig. 25). The maximal relative difference on G along the crack front is less than 0.6%, which validates the optimal mesh refinement procedure proposed in this paper.

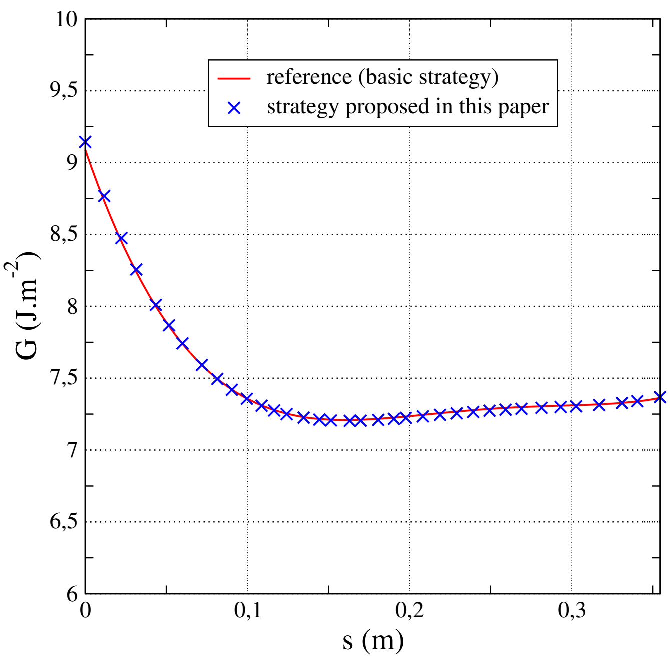


**Fig. 25.** Evolution of the energy relase rate G along the crack.

## 6. Conclusion

In this paper, a method for the conformal mesh refinement if the initial mesh is made of hexahedra has been presented. The transition between two zones with a different level of refinement is made by a combination of pyramids and tetrahedra. A set of patterns was designed to solve any conflict about the pending nodes on the faces of the hexahedra. From the first method we presented before, we made an important improvement that minimizes the propagation of the refinement. Thanks to this new method, we can apply the mesh adaptation to every structure, whatever the type of the elements of the initial mesh.

## References


[1] Frey P-J, George P-L. Mesh generation. Applications to finite element methods. Wiley; 2008.
[2] Zienkiewicz OC, Zhu JZ. A simple error estimator and adaptive procedure for practical engineering analysis. Int J Numer Meth Eng 1987;24.
[3] Prudhomme S, Oden JT. On goal-oriented error estimation for elliptic problems: applications to the control of pointwise errors. Comput Methods Appl Mech Eng 1999;176:313–31.
[4] Ern A, Meunier S. A posteriori error analysis of Euler-Galerkin approximations to coupled elliptic-parabolic problems. ESAIM: M2AN 2009;43:353–75.
[5] Frey P-J, Alauzet F. Anisotropic mesh adaptation for CFD computations. Comput Methods Appl Mech Eng 2005;194:5068–82. doi:10.1016/j.cma.2004.11.025.
[6] Ruiz-Girones E, Roca X, Sarrate J, Montenegro R, Escobar JM. Simultaneous untangling and smoothing of quadrilateral and hexahedral meshes using an object-oriented framework. Adv Eng Softw 2015;80:12–24. doi:10.1016/j.advengsoft.2014.09.021.
[7] Kowalski N, Ledoux F, Frey P. Block-Structured Hexahedral Meshes for CAD Models using 3D Frame Fields. In: Proceedings of the 23rd International Meshing Roundtable; 2014.
[8] Tautges TJ. The generation of hexahedral meshes for assembly geometry: survey and progress. Int J Numer Meth Eng 2001;50:2617–42. doi:10.1002/nme.139.
[9] Huang L, Zhao G, Ma X, Wang Z. Incorporating improved refinement techniques for a grid-based geometrically-adaptive hexahedral mesh generation algorithm. Adv Eng Softw 2013;64:20–32. doi:10.1016/j.advengsoft.2013.04.010.
[10] International Meshing Roundtable, http://imr.sandia.gov/papers/mesh.html.
[11] Bank RE. PLTMG, a software package for solving elliptic partial differential equations, Philadelphia: SIAM; 1990. User's guide 6.0".
[12] Bastian P, Birken K, Johannsen K, Lang S, Neuss N, Rentz-Reichert H. UG – A flexible software toolbox for solving partial differential equations. Comput Vizual Sci July 1997;1(1).
[13] Rivara M-C. Local modification of meshes for adaptive and/or multigrid finite element methods. J Comp Appl Math 1991;36:79–89.
[14] Sohn D, Lim JH, Im S. An efficient scheme for coupling dissimilar hexahedral meshes with the aid of variable-node transition elements. Adv Eng Softw 2013;65:200–15. doi:10.1016/j.advengsoft.2013.06.017.
[15] Kim HG. Development of three-dimensional interface elements for coupling of non-matching hexahedral meshes. Comput Methods Appl Mech Eng 2008;197:3870–82. doi:10.1016/j.cma.2008.03.023.
[16] Li Y, Liu GR, Zhang GY. An adaptive NS/ES-FEM approach for 2D contact problems using triangular elements. Finite Elem Anal Des 2011;47:256–75. doi:10.1016/j.finel.2010.10.007.
[17] Lo SH, Wu D, Sze KY. Adaptive meshing and analysis using transitional quadrilateral and hexahedral elements. Finite Elem Anal Des 2010;46:2–16. doi:10.1016/j.finel.2009.06.010.
[18] Owen SJ, Saigal S. Formation of pyramid elements for hexahedra to tetrahedral transitions. Comput Methods Appl Mech Eng 2001;190:4505–18.
[19] Nicolas G, Fouquet T. Adaptive mesh refinement for conformal hexahedral meshes. Finite Elem Anal Des 2013;67:1–12. doi:10.1016/j.finel.2012.11.008.
[20] *HOMARD* web site: / www.code-aster.org/outils/index.en.html or www.salome-platform.org.
[21] Geniaut S, Galenne E. A simple method for crack growth in mixed mode with X-FEM. Int J Solids Struct 2012;49:2094–106. doi:10.1016/j.ijsolstr.2012.04.015.
[22] Colombo D, Massin P. Fast and robust level set update for 3D non-planar X-FEM crack propagation modelling. Comput Methods Appl Mech Eng 2011;200:2160–80. doi:10.1016/j.cma.2011.03.014.
[23] *Code_Aster* Finite Element software for mechanical issues by EDF, www.code-aster.org.