**Proceedings of the ASME 2009 International Design Engineering Technical Conferences &
Computers and Information in Engineering Conference
IDETC/CIE 2009
August 30 – September 2, 2009, San Diego, CA, USA**

DETC2009-86575

TOWARDS CAD-LESS FINITE ELEMENT ANALYSIS USING GROUP BOUNDARIES FOR ENRICHED MESHES MANIPULATION

R. Lou[⊥,*,◊], F. Giannini*, J-P. Pernot[⊥], A. Mikchevitch[◊], B. Falcidieno*, P. Véron[⊥], R. Marc[◊]

[⊥]LSIS – UMR CNRS 6168
2 cours des Arts et Métiers
13617 Aix-en-Provence, France
{ruding.lou, jean-philippe.pernot, philippe.veron}@ensam.fr

***IMATI – CNR**
6, via de Marini
16149 Genova, Italy
{falcidieno,giannini}@ge.imati.cnr.it

[◊]Electricité de France Group
Research and Development Direction
1, avenue Général de Gaulle
92141 Clamart cedex, France
{alexei.mikchevitch, raphael.marc}@edf.fr

ABSTRACT

Nowadays, most of the numerical simulations are carried out by successively performing the following steps: CAD model definition or modification, conversion to a mesh model and enrichment with semantic data relative to the simulation (e.g. material behaviour laws, boundary conditions), Finite Element simulation and analysis of the results. Classically, the semantic data are attached to the mesh through the use of groups of geometric entities sharing the same characteristics. Thus, any modification of the CAD model always implies an update of the mesh as well as an update of the attached semantic data. This is time-consuming and not adapted to the context of industrial maintenance. Moreover, the CAD models do not always exist and should therefore be reconstructed starting from scratch or from the physical object. In this paper, we set up a framework towards the definition of CAD-less Finite Element analyses wherein enriched meshes are manipulated directly. The geometric manipulations are constrained with information extracted from the group definition. Actually, the boundaries of those groups are exploited to constrain the modifications. The concept of Virtual Group Boundaries is introduced to focus on the extension of the attached semantic information instead of the actual tessellation while generalising the approach to groups of any dimension going from 0D (vertex) to 3D (e.g. tetrahedron). The notion of Elementary Group is also introduced as a mean to ease the forthcoming transfer of the semantics from the initial to the modified models. Such a framework also finds interest in the preliminary design phases where alternative solutions have to be evaluated.

1. INTRODUCTION

Today, the mainstream methodology for product behaviour numerical simulations relies on the following steps: conceptual phase, Computer-Aided Design (CAD) modeling, meshing and model preparation for specific behavior study, Finite Element (FE) simulation, result analysis and optimization loops [1,2]. Such a process is illustrated in figure 1 wherein dot lines show the general workflow when performing successive optimizations. For each modification, a come back to CAD modeling is required that implies an updating of the mesh as well as some adjustments in all the forthcoming steps. This is time-consuming. Actually, in such a process, most of the time is spent for the development of complex meshes adapted to specific FE simulations, for the accurate identification of the unknown parameters and for the prototyping and assessment of the proposed solution. Thus, it would be important to preserve as much as possible all the manipulations and data that have been previously set up. This is especially true when the simulation models have been tuned to fit the ground truth that can be measured on the real object.

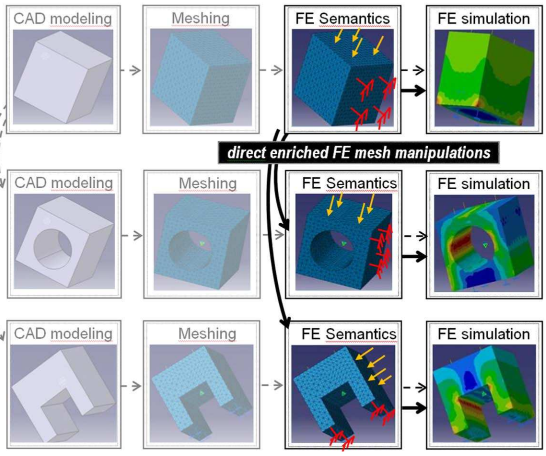


Figure 1: Classical design process (dot lines) *vs.* CAD-less design process adapted to the fast industrial maintenance analysis and preliminary design phases (continuous lines) [3].

There is therefore an easy understandable need for developing fast prototyping methods and tools applied to the maintenance context. As a reference, for the Electricité de France (EDF) Group, the total cost of one day of stop of a nuclear power plant represents several hundreds of thousands of euros [3]. Moreover, in the context of industrial maintenance and life-cycle problem analysis, the product is already designed, the CAD models are not necessarily available and the product behavior has to be studied and improved during its exploitation. Therefore, it seems promising to go towards the definition of a CAD-less FE analysis methodology acting directly at the level of the FE mesh enriched with semantics relative to the simulation models like material behavior laws, boundary conditions (BCs), geometric/mechanical parameters, etc. The continuous line of figure 1 show how a CAD-less approach would skip the CAD modeling and meshing steps to directly work at the level of the enriched meshes.

When thinking to a CAD-less methodology for fast modifications and prototyping of alternate solutions, not only some geometric modifications of the meshes have to be considered, but also the treatment of the semantics (e.g. material behavior laws, boundary conditions) relative to the FE simulation model. Classically, the semantic data are attached to the mesh through the use of groups of geometric entities sharing the same characteristics. For example, the specification of a pressure on the outer surface of a model will involve a group of faces. Similarly, the definition of the material constituting a tetrahedral mesh will require a group of tetrahedral elements. To be able to maintain and/or to transfer this information during the mesh modification, the shape (boundaries) of the groups have to be considered. Indeed, they constrain the geometric modifications and it is therefore crucial to be able to identify and take them into account when doing any mesh modifications.

In this paper, the concept of Virtual Group Boundaries is introduced to set up a generic approach for manipulation of groups of any dimension going from 0D (vertex) to 3D (e.g. tetrahedron). The notion of Elementary Group is also introduced as a mean to ease the forthcoming transfer of the semantics from the initial to the modified models. The paper is organized as follows. Section 2 comes back on the needs for fast prototyping tools while introducing complex examples by EDF. The proposed framework is presented in section 3 and detailed in section 4, namely some aspects of group boundary handling are discussed. Some results of such a handling coming from our prototype software are presented in section 5.

2. NEEDS FOR FAST PROTOTYPING TOOLS

To further illustrate the needs for setting up a fast prototyping framework with dedicated tools, let us consider the example of figure 2 presenting a complex case the EDF engineers have to deal with. Figure 2.a shows the CAD model of a "u-like" testing bench in which a structural modification has to be performed (addition of stiffeners in the present case). According to the traditional prototyping workflow, this study would include the following steps:

1) development of the complex CAD model which does not exist (step ❶);
2) advanced meshing satisfying mesh quality criteria, mechanical modeling hypotheses and so on (step ❷);
3) creation of numerous mesh entity groups that will support the semantics defined in step (5). Here, there are 35 groups (17 groups of nodes, 10 of faces and 8 of tetrahedral entities). They are created either manually while selecting a set of mesh entities or semi-automatically while selecting partitions of the CAD model. This step requires a very good skill and is time-consuming (step ❸);
4) enrichment of the FE model with semantics and modeling hypotheses (step ❸):
    - describing mechanical links between model partitions geometrically separated,
    - characterizing several materials differenced by colors on figure,
    - describing specific geometrical/mechanical parameters (beam modeling, spring discrete element modeling, punctual mass tuned),
    - defining BCs and different loads;
5) tuning of the FE model through experimental results;
6) FE analysis properly saying (step ❹);
7) prototyping of the envisaged solution (addition of stiffeners in the present example) through an update of the initial CAD model (step ❺);
8) preparing new mesh model for FE simulation (steps ❻ to ❼) and going back to the FE analysis step to evaluate the proposed solution (step ❽), and so on.

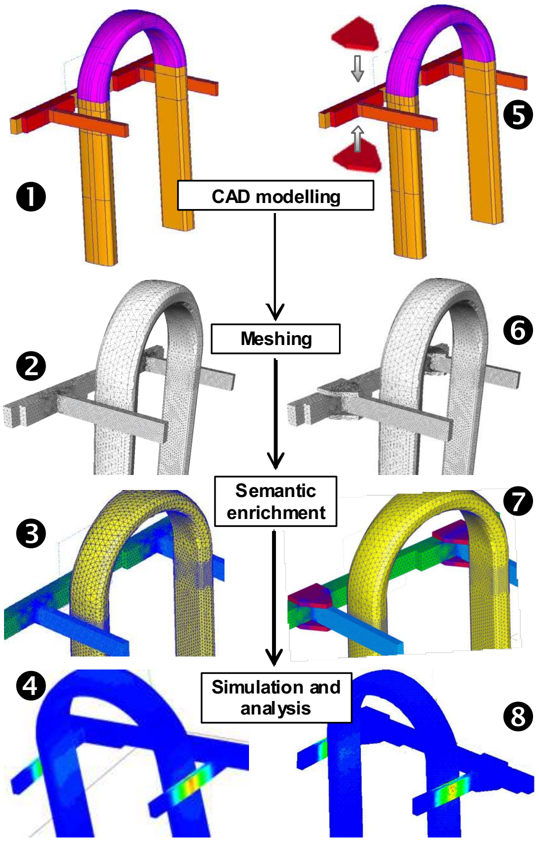
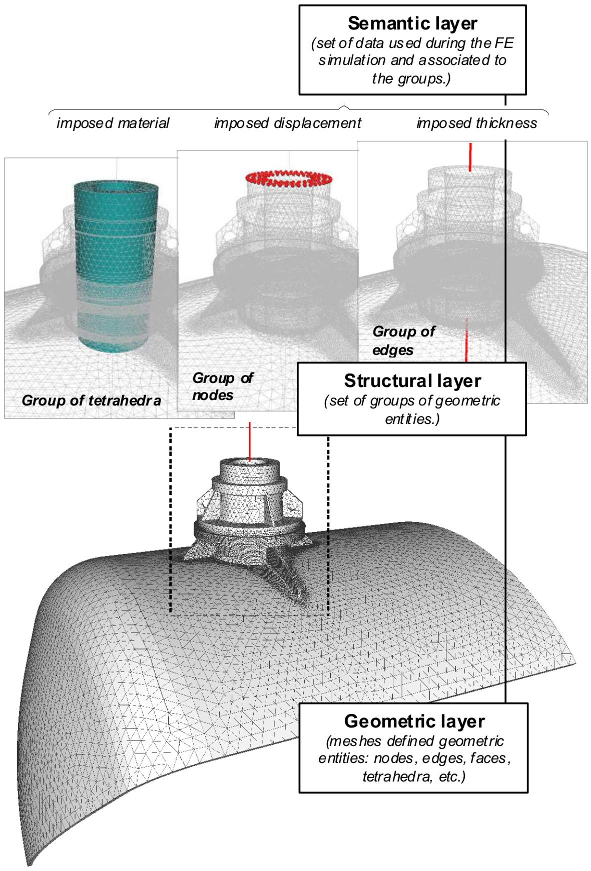


Figure 2. Example of process pipeline for FE simulation on an industrial case, where after an initial simulation (1-4) two stiffeners are added requiring performing the complete CAO-Meshing-FE Analysis process (5-8) (courtesy EDF R&D).

Here, it seems quite clear that going back to the CAD model is not the most appropriate solution. This is especially true when the model contains a huge amount of semantic data. As a reference, some EDF models may contain up to 500 mesh groups dedicated to the FE analysis (BCs, link relations, various behavior laws, geometric parameters, mechanical modeling of specific phenomena, etc.) as well as to the post-processing. Unfortunately, current CAD software (I-DEAS/NX®, CATIA®, SALOME®, etc.) do not make it possible to fully automatize this complex process. Thus, the prototyping and optimization of structural modifications may require several time-consuming complete updates of the numerical model. This is not acceptable for fast industrial studies. Moreover, there exists another limit when using CAD models as inputs of a FE simulation. Actually, CAD models mainly consider the outer surfaces of the object as perfect and do not take into account the real shapes that can be measured on-site together with its imperfections. When simulating real structures, the meshes should always been tuned to better fit the shapes and behaviors of what can be measured (e.g. lasers) on the real installation. For example, the shapes of the mesh of figure 2 are closer to the reality than the exact surfaces (e.g. cylinders, plans) used to represent them into the CAD model. Somehow, the meshes take into account more imperfections when they have been tuned according to the physical product. They are more adapted to design "as-built" models. Thus, going towards the definition of a CAD-less FE analysis framework seems promising.

Semantic aspects have been widely detailed and exemplified in the Aim@Shape European Network of Excellence [4]. In industrial design, the semantic data correspond to all the information that is used to design and manufacture a product: its colors, its material, its decomposition into meaningful areas. In the context of FE simulation, they may also correspond to all the data required before running the simulation properly saying: BCs, geometric parameters (thickness of a group of faces), materials and so on [5]. Actually, semantic aspects can be encountered in all the steps of the product lifecycle. Integration and maintenance of semantic information during the product modeling process has been subject of research since many years [6,7]. Some approaches try to take into account this application-dependent information during the manipulation and modification of the underlying geometric models. Hamri et al. [8,9] have proposed an unified framework to handle and to process the CAD models and FE meshes through an intermediate polyhedral representation. In their approach, the semantic data are taken into account through the specification of partitions whose boundaries may drive a polyhedral simplification method used to adapt the digital mock-up to the various engineering needs (e.g. visualization, FE simulation, clash detection). This is an interesting example on how semantic information can constrain geometric manipulations.

However, to the best of our knowledge, the preservation and propagation of simulation semantics during the mesh manipulation has not been addressed yet. Most of the existing methods and software treat the problem on a geometric point of view while forgetting that the geometry is often used as a support used to convey semantic information all along the design process. Hence, a specific effort has to be done in this direction.



Semantic layer
(set of data used during the FE simulation and associated to the groups.)
imposed material
imposed displacement
imposed thickness
Group of tetrahedra
Group of nodes
Group of edges
Structural layer
(set of groups of geometric entities.)
Geometric layer
(meshes defined geometric entities: nodes, edges, faces, tetrahedra, etc.)

Figure 3. Geometric, structural and semantic layers taking part to the definition of a FE simulation model (courtesy EDF R&D).

3. A MULTI-LAYERED MODULAR APPROACH

Classically, in a FE model, it is possible to identify different layers of information: a semantic, a structural and a geometric one. The latter describes the shape of the product by mean of discrete geometric models characterized by some specific properties, e.g. regularity and conformity. At this level, there exist all the categories of meshes (e.g. beam, triangle, quadrangle, hexahedral or tetrahedral meshes) including mixed and non-manifold ones. The semantic layer gathers together the geometric parameters (cross-section properties, thickness), the mechanical parameters (behavior laws, stiffness) as well as the conditions to which the product is submitted (e.g. imposed displacement or forces, link relationships, contact conditions) to allow a simulation reflecting as much as possible the real behavior of the modeled structure. This semantic information refers to some parts of the structure and is then included in the model by referring to the geometric elements representing the affected parts. In most of the commercial software, this is done through the concept of groups, which collect all the geometric elements concerned by specific semantic information. At the end, FE meshes are enriched with semantic data through the use of a set of FE entity groups structuring somehow the geometric models. Figure 3 shows an example of such a multi-layered decomposition when considering the FE simulation of one-height of a cylindrical caisson. On more complex industrial examples, there can be up to hundreds of FE mesh groups defined over a single mesh. In this case, the set of groups takes part to the definition of several distinct simulation models that can be run using a unique geometric model on which different simulation semantics can be assigned and so, various FE simulations can be accomplished.

Thus, mesh modifications not only have to consider the geometric layer but also the structural and semantic ones. Indeed, correctly propagating the already specified semantic information during the shape modification means not only to propagate or to maintain the group structure over the new mesh but also to reason on the specific type of associated semantics. This is due to the fact that the transfer of the semantic data influences the shape modification operator itself according to these different layers of information. At the lowest level, i.e. the geometric layer, the operator modifies the shape by adding, removing or even deforming geometric elements. The second level makes use of the groups and corresponds to the mesh modification under constraints (group topology, position and shape). Finally, at the highest level the nature of semantics associated to the groups for FE simulation specifies if and how the groups have to be propagated and updated.

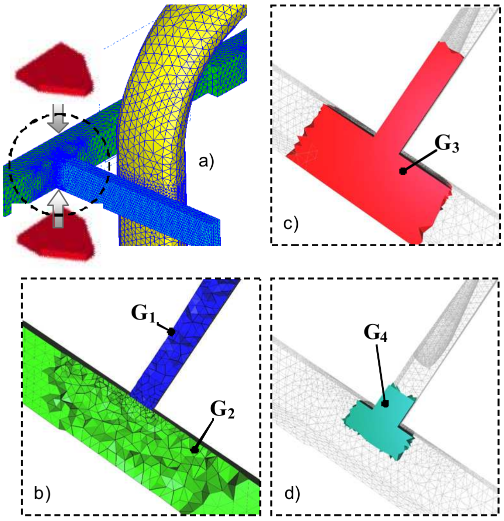


Figure 4. Overlapping FE mesh groups (b, c, d) at contact and welding zone of the beams (a) of the "u-like" testing bench model.

Moreover, on real industrial examples, several semantic data can affect the same areas of a mesh. Thus, several groups may refer to the same geometric mesh elements. Figure 4 shows an example of overlapping groups on the example of the "u-like" testing bench model already discussed in the previous section. Around the beam assembly zone (contact and welding zone) of figure 4.a, four groups G_i of tetrahedral elements interact (fig. 4.b to 4.d). These 3D groups overlap each other at the contact zone of the beams. Therefore, adding a stiffener at this zone needs to consider all these overlapping groups. To efficiently deal with such overlapping configurations, we have introduced the notion of elementary group and associated manipulation operators. The next section details these aspects as well as the way group boundaries can be used to constrain required geometric modifications of the FE mesh model. An elementary group is the maximal subset of mesh elements belonging to the same set of groups. Conversely a group may contain several elementary groups. By doing this way, we reach a high level of structuring. We consider not only a set of groups but also their mutual interactions are taken into account. This is a key point to be able to drive efficiently the mesh modifications as well as the transfers of corresponding semantics.

4. ON THE USE OF GROUP BOUNDARIES

The needs for taking into account overlapping groups during the mesh modifications have been presented in the previous section. This section details how these groups and the information they support can be handled. More precisely, the notions of group boundaries and elementary non-overlapping groups are introduced as a mean to constrain required the mesh modifications as well as to help the propagation of the semantic data in the modified areas.

4.1. Presentation of merging process on example of triangle meshes

To be able to merge triangle meshes of widely varying densities while satisfying mesh quality criteria in the intersection area, a specific algorithm has been developed [3]. This method is illustrated on the example of figure 5 corresponding to the fuse problem of two mesh models. The first step of merging process consists in computing the intersection line and optimizing the number and position of it nodes (fig. 5.b). The contact zone between the two parts is then cleaned by removing all the triangles in a bandwidth around the intersection line (fig. 5.c). A transition area is then created to enable a smooth transition between the two triangle mesh densities.

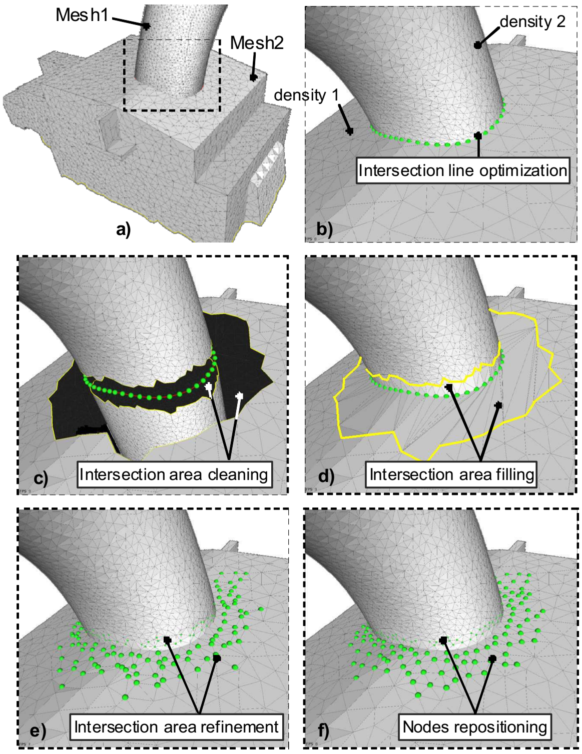

Mesh1
Mesh2
density 1
a)
density 2
Intersection line optimization
b)
c)
Intersection area cleaning
d)
Intersection area filling
e)
Intersection area refinement
f)
Nodes repositioning

Figure 5. Steps of the triangle mesh merging algorithm.

The holes thus created are then triangulated (fig. 5.d) and refined (fig. 5.e) to make the size of the new triangles compatible with the original surrounding mesh entities. Finally, the quality of the new triangles is improved by applying a deformation technique based on the Force Density Method [10] that allows making triangles as much equilateral as possible (fig. 5.f).

According to us, the transfer of semantic information during the mesh modification inevitably has to go through the preservation/propagation of the group boundaries. As illustrated on the example of figure 6, when considering the intersection of two triangle meshes on which FE entity groups have been defined (fig. 6.a), it is mandatory to preserve their boundaries when handling the intersection areas. Here, the boundaries of the groups are preserved, i.e. they are not deleted, during the cleaning operation (fig. 6.b). Thus, the shape of face groups is preserved and the semantics attached to these groups in the cleaned areas can be accurately reassigned.

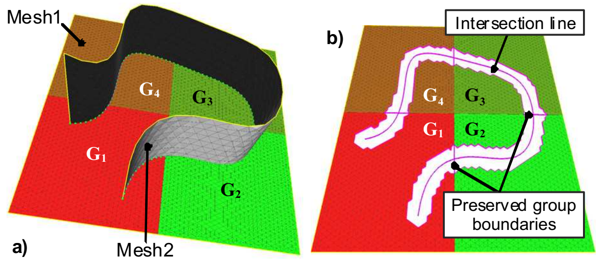


Figure 6. Mesh merging with four groups whose boundaries are preserved during the mesh modification process.

4.2. Concept of Virtual Group Boundaries (VGB)

Since the boundary of the group does not correspond precisely to the topological notion of boundary, it requires a specific definition. This is even truer when the boundary does not exist geometrically (in the case of node group, for example) and should be therefore evaluated. Thus, we introduce the so-called Virtual Group Boundary (VGB) as a set of 0D, 1D and/or 2D elements located at the group limit. The dimension of FE entities constituting the group boundary depends on the dimension of the mesh (d_M) as well as on the dimension of the elements constituting the group (d_G). For a given group, the VGB can be decomposed in two potentially empty subsets:

- a **set of Bounding Elements (BE)** gathering together the elements of the mesh having a dimension (d_M -1). They may not belong to the group but constitute one or more connected sets enclosing portions of the mesh whose elements of dimension d_G belong to the group,
- a **set of Isolated Elements (IE)** belonging to the group and having a dimension d_G.

The reason why we consider these two sets to define the VGB is motivated by an impact of their constitutive elements (their topology and position) on mesh modification process under constraints in order to propagate accurately the group data in the resulting mesh after modification.

4.3. VGB specification

In order to compute VGBs that can be identified from groups in the case of 2D/3D meshes, we have been considering an exhaustive list of possible configurations.

4.3.1. VGB of groups defined over a 2D mesh

Depending on the dimension of the elements constituting the considered group, three configurations are distinguished and illustrated on figure 7:

- the **VGB of a group of 2D elements** (faces) belonging to a 2D mesh is defined by one set of BE gathering together some of the edges of the mesh. These edges are associated to only one face *f* of the group so that *f* has at least one edge in common with another face of the group. They define closed loops enclosing one or several connected faces of the 2D group. Figure 7.a$_1$ shows a group of faces whose VGB is identified in figure 7.a$_2$.
- the **VGB of a group of 1D elements** (edges) belonging to a 2D mesh is defined by two sets of edges. The set of BE gathers together the edges associated to exactly one face *f* whose edges are in the group and so that *f* has at least one edge not considered as BE. They form closed loops that enclose mesh areas in which all the edges are components of the 1D group. The set of IE contains the edges associated to no face on which all the edges are in the group or to faces in which all the edges may be classified as BE. In this case, both the BE and IE belong to the group. Figures 7.b$_i$ show a group of edges and the corresponding VGB made of BE and IE. On this example, the transfer of the three bounding edges belonging to the same face from the set of BE to the set of IE prevents a wrong semantic data transfer due to shape tessellation. Actually, nothing guarantees that new edges inserted in the concerned face have to be included into the group.
- the **VGB of a group of 0D elements** (nodes) belonging to a 2D mesh is constituted by a set of edges and a set of nodes. The set of BE gathers together the edges associated to only one face *f* whose nodes (all) are in the group and such that *f* has at least one edge not BE. The set of IE contains the nodes associated to faces having only one or two nodes in the group or to faces in which all the edges may be classified as BE. This case is illustrated on figures 7.c$_i$. Here a transfer of the three nodes associated to the three bounding edges from the set of BE to the set of IE prevents a wrong semantic data transfer due to shape tessellation (encircled nodes on fig. 7.c$_2$).

Analogical classification and definition of VGB can be used in the case of quadrangular meshes.

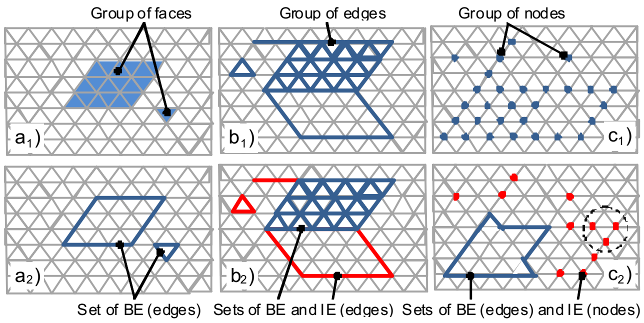


Figure 7. Examples of VGBs extracted from groups defined over a 2D triangle mesh.

4.3.2. VGB of groups defined over a 3D mesh

Similarly, depending on the dimension of the elements constituting the group, four configurations can be distinguished:

- the **VGB of a group of 3D elements** (e.g. tetrahedral elements) belonging to a 3D mesh is formed by a set of BE corresponding to faces associated to only one 3D element of the group. The set of IE is empty. Here, the BE may also contain all the faces enclosing the same 3D element. This is due to the fact that the associated semantic information is relative to the enclosed volume and it is therefore meaningful to propagate it to new 3D elements that might be inserted during the mesh modifications. This won't be true for all the following configurations (2D to 0D elements) since in these cases corresponding to more "discrete" information, the BE belonging to the same 3D element are considered as IE thus expressing somehow the uncertainty about the semantic information propagation.

- the **VGB of a group of 2D elements** (faces) belonging to a 3D mesh is constituted by two sets of faces. The set of BE gathers together the faces associated to only one 3D element for which all the associated faces are in the group and so that not all its faces are classified as BE. The set of IE gathers together the faces associated to no 3D element on which all the associated faces are in the group or to 3D elements for which all the faces may be classified as BE.

- the **VGB of a group of 1D elements** (edges) belonging to a 3D mesh is constituted by a set of faces and a set of edges. The set of BE contains the faces associated to only one 3D element on which all the associated edges are in the group and so that not all its bounding faces can be classified as BE. The set of IE gathers together the edges associate to no 3D element on which all the edges are in the group or to 3D elements in which all the associated faces can be classified as BE.

- The **VGB of a group of 0D elements** (nodes) belonging to a 3D mesh can be constituted by a set of faces and a set of nodes. The set of BE gathers together the faces associated to only one 3D element for which all the nodes are in the group and so that all its faces cannot be classified as BE. The set of IE contains the nodes relative to no 3D element on which all the nodes are in the group or to 3D elements in which all the associated faces can be classified as BE.

The transfer of the bounding faces belonging to the same 3D element from the set of BE to the set of IE will prevent a further decomposition of the 3D element.

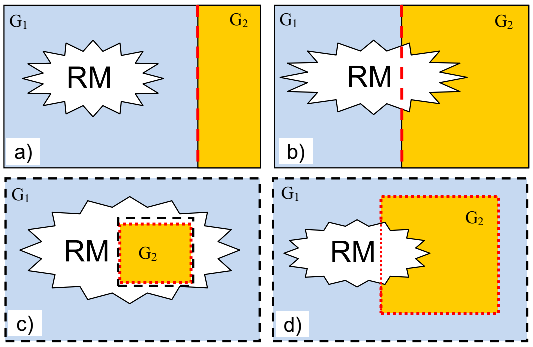


Figure 8. Configurations of re-meshing zone (RM) interacting with overlapping groups (G_1, G_2).

4.4. Interaction between group and remeshing area

This section presents various interactions between the re-meshing zone (see section 4.1) and the VGB. It aims at introducing algorithms to handle the problem of group propagation from the initial elements to the newly created ones.

- **Re-meshing zone fully inside the VGB (fig. 8.a)**: in this case, the assignment of the group information to the re-meshed part can be performed directly by considering the group specified for neighbor elements of the re-meshing area. Figure 8.a shows an example of this case. The model contains two groups G_1 and G_2, the re-meshed zone (RM) is inside G_1. After re-meshing step, all the new elements in RM have to inherit the group information from the other side of the RM boundary.

- **Re-meshing zone intersecting the VGB (fig. 8.b)**: in this case, RM should respect the intersected VGB and should therefore be divided into sub-RMs. Each sub-RMs should reference to the group definition of neighbor element of RM boundary. Figure 8.b shows an example in which the RM is intersecting the VGBs of two groups G_1 and G_2. At first, the re-mesh operation should respect the red dashed line that is a part of the VGB intersecting with RM. The RM is then divided into left and right parts. Finally, the two parts can inherit the group information from other sides of the RM boundary, i.e. G_1 and G_2 respectively.

- **Re-meshing zone enclosing the VGB (fig. 8.c):** RM can also include completely a VGB or a disconnected closed sub-VGB. In this case the method used in the previous configurations cannot be applied since all the elements inside the included VGB (or sub-VGB) are removed. A possible solution could be to assign a group "label" to the VGB. After geometric mesh modification, every new element should get group information from the surrounding VGB. An example is given in figure 8.c in which the VGBs of the two groups G_1 and G_2 are overlapping (internal black dashed line and red dashed line are overlapping). The group G_2 is completely inside of the RM. At first, the re-meshing operation should maintain the VGB of G_2. Then all new elements of RM should find the surrounded VGB to get the information. The elements inside the VGB of G_2 should be assigned as G_2 and the elements surrounded by the VGB of G_1 should be updated as G_1. Here, the VGB of G_1 is composed by two disconnected edge loops: one external to the re-meshing area and one internal, i.e. the one overlapping with the VGB of G_2.

- **Re-meshing zone interacting with overlapping groups (fig. 8.d):** it concerns the case in which different groups contain the same elements, i.e. one FE entity can be assigned to more than one group. Figure 8.f shows an example in which all the elements of the group G_2 belong also to G_1. RM corresponds to elements belonging only to G_1 and elements belonging to both G_1 and G_2 at the same time. At first, the re-meshing operation should be applied taking into account the VGB of G_1 and G_2. Then every VGB (either of G_1 and G_2) should give the group definition to all bounded elements. In the RM zone, the elements at right of the red dashed line segment should be given a double group definition respectively from the VGB of G_1 and G_2. The elements at left of the red dashed line segment will just be assigned with G_1.

In the case of the two last configurations, the new inserted elements are bounded by more than one group boundary and the re-assignment to a group for these new elements is not determinist. To overcome such a problem, the notion of **Elementary Group (EG)** has been introduced as a mean to split the semantic groups into non partially overlapping parts.

4.5. Decomposition into Elementary Groups (EG)

The Elementary Groups can be defined as sub-groups containing the same FE entities of a given topology belonging to different groups of the FE mesh model. This concept appears at the structural level, between the semantic and geometric levels (section 3). Thus, the subdivision into EGs will neither affect the geometric entities nor the semantic data. Actually, EGs are created from original groups to avoid partially overlapping configurations, except for elements on the VGB. To sum up, all mesh entities belonging to an elementary group and which are not on the VGB cannot partially belong to another elementary group. Moreover, EGs deal also with groups constituted by geometric elements of different dimensionality (e.g. nodes, edges, faces, 3D elements) that spatially overlap to guarantee the simultaneous setting of constraints during the shape modification.

• INT	elem_dim	(node/edge/face/3D element)
• ARRAY	domain	(mesh elements of ***d_m*** for representing group space)
• ARRAY	bounding	(VGB's bounding elements of dimension ***d_m -1***)
• ARRAY	isolated	(VGB's isolated elements)
• GROUP	group	(associated FE group)
• ARRAY	entities	(mesh entities in this elementary group)

Figure 9. EG data structure.

The data structure to handle the EGs is summarized in the array of figure 9. The integer "elem_dim" indicates the dimension of the elements contained in the EG. The field "domain" gathers together all the elements of dimension d_M (mesh dimension) covered by the entities of the EG. For example, when an EG of a 2D mesh contains a set of nodes, the field "domain" gathers together the underlying faces to which the nodes belong. When an EG of a 3D mesh contains a close set of faces, the field "domain" gathers together the 3D elements bounded by this set of faces. The "bounding" and "isolated" fields respectively contains the BE and IE of the VGB. The field "group" refers to the FE group from which this EG was created. The "entities" corresponds to those elements forming the EG extracted from the FE group. Actually, BEs enclose the space of dimension d_M defined by elements included in the field "domain". IEs can be considered as punctual elements, which cannot form a space of dimension d_M.

```
$DOM_1 \leftarrow$ elements in "domain" of $G_1$
$DOM_2 \leftarrow$ elements in "domain" of $G_2$
if ($DOM_1 \neq DOM_2$)
 $DOM_3 \leftarrow DOM_1 \cap DOM_2$
 if ($DOM_3 \neq \emptyset$)
   $DOM_1 \leftarrow DOM_1 - DOM_3$
   $DOM_2 \leftarrow DOM_2 - DOM_3$
   if ($DOM_1 \neq \emptyset$)
     Create elementary group $G_3$
     For each $e$ (entity in $G_1$)
       if ($e$ associates to elements in $DOM_3$)
         Add $e$ in $G_3$
         if ($e$ do not associates to any elements in $DOM_1$)
           Remove $e$ from $G_1$
         End if
       End if
     End for
   End if
   if ($DOM_2 \neq \emptyset$)
     Create elementary group $G_4$
     For each $e$ (entity in $G_2$)
       if ($e$ associates to elements in $DOM_3$)
         Add $e$ in $G_4$
         if ($e$ do not associates to any elements in $DOM_2$)
           Remove $e$ from $G_2$
          End if
       End if
     End for
   End if
   For $G_1, G_2, G_3$ and $G_4$
     Update the fields "domain", "bounding",
     "entities", "group", "elem_dim"
   End for
End if
```

Figure 10. The EGs splitting algorithm.

The main steps for creating EGs from FE groups are detailed hereafter:

- first, for each FE group containing elements of the same dimension, a corresponding EG is created so that the field “entities” contains all the elements of the FE group. At the opposite, each FE group that contains elements of different dimensions will be decomposed into several EGs whose fields “entities” contain elements of the same dimension;

- then, the field “domain” is update by looking for the underlying faces of a 2D mesh or the underlying 3D elements of a 3D mesh. The “bounding” field is updated by looking for either all the faces associated to only one 3D element of the “domain” for a 3D mesh, or all the edges associated to more than one face of the “domain” and to more than one face not in the “domain” for a 2D mesh. The “isolated” field gathers together all the elements contained in the field “entities” and which are not associated to elements of the “domain”;
- finally, a set of operations are performed between all the couples of EGs to produce potentially new non partially overlapping EGs. These computations use the algebra of 2D and 3D sets to evaluate the relationships between the initial EGs. When partially overlapping configurations are detected, the initial EGs are split. The figure 10 details the algorithm used to produced those EGs from two potentially partially overlapping EGs G_1 and G_2.

Figure 11 illustrates the complete algorithm on example of two partially overlapping FE groups. The process starts from two Elementary Groups EG_1 and EG_2 (fig. 11.a) directly computed from two FE groups of dimensions 2 (group of faces) and 0 (group of nodes). Each EG is then treated to identify the elements belonging to the fields “domain”, “bounding” and “isolated” (fig. 11.b_i and 11.c_i). The intersections of the two initial “domains” are then computed (fig. 11.d_i) to enable the definition of four non partially overlapping EGs (fig. 11.e_i).

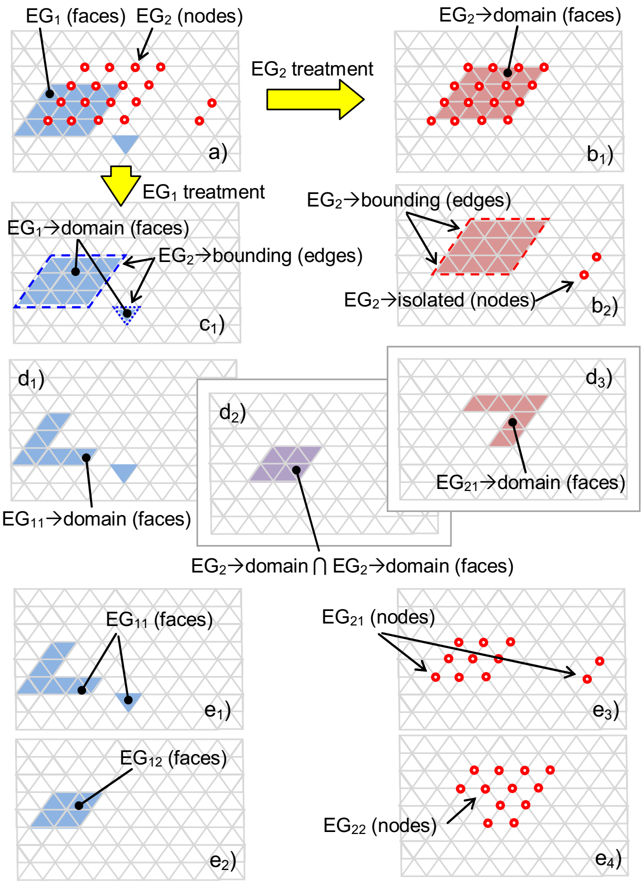


Figure 11. Definition of EGs from two partially overlapping groups of nodes and faces.

5. RESULTS

To illustrate how EGs, and their corresponding VGBs, can be computed from an initial set of FE groups, we consider the industrial example of figure 12.a corresponding to a 2D triangle mesh of a caisson involved in a fast operational study of EDF. At the geometric level, the mesh is defined by 38672 nodes, 115253 edges and 76582 triangles. At the structural level, the model is structured with 5 groups of nodes and 8 groups of faces that partially overlap. Figures 12.b and 12.c show two groups of nodes whereas figures 12.d, 12.e and 12.f show three groups of faces.

The algorithm discussed in section 4.5 produces 34 EGs. Figures 13.a_1, 13.a_2 and 13.a_3 show the EGs relative to the node group depicted on figure 12.b: initial node group has been split in three EGs of nodes. Figures 13.b_1, 13.b_2 and 13.b_3 show three of the eight EGs computed from face group shown in figure 12.e. On figures 13.b_i, the green edges correspond to the BEs of each EG. Theses edges will be used as constraints during the mesh modification process.

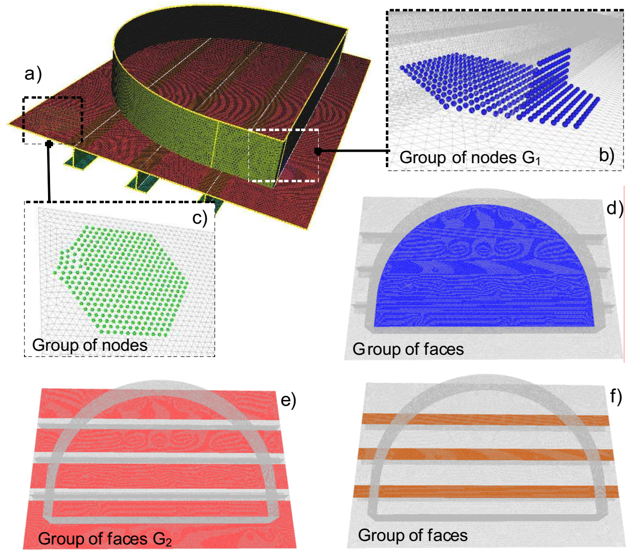


Figure 12. Examples of partially overlapping groups on the 2D triangle mesh model of the caisson (courtesy EDF R&D)

6. CONCLUSION AND FUTURE WORK

This paper sets up the bases for the definition of a CAD-less FE analysis framework wherein enriched FE mesh models are directly manipulated, i.e. without going back to the CAD models. The concepts of Elementary Groups (EGs) and associated Virtual Group Boundaries (VGBs) are introduced as a mean to help the preservation and propagation of the semantic data during the mesh modification operations. VGBs can be defined for EGs of any dimension going from 0D mesh entities (nodes) to 3D elements. The approach acts at three levels: geometric, structural and semantic layers. In this paper, mainly the operations performed at the structural level have been detailed and illustrated through examples coming from our prototype software.

The next step concerns the use of these VGBs as inputs for mesh modification process, for example, for mesh fuse problem presented in section 4.1. Actually, the idea is to use the VGBs as constraints for re-meshing and deformation operators during the mesh modification operations (adding/removing of material, cutting of structure). VGBs can be defined over 2D or 3D meshes and the development of the 3D case is actually in progress. The decomposition of FE groups into EGs allows to propagate accurately the semantic data as well as to update correctly mesh groups required for FE analysis after fast mesh modification.

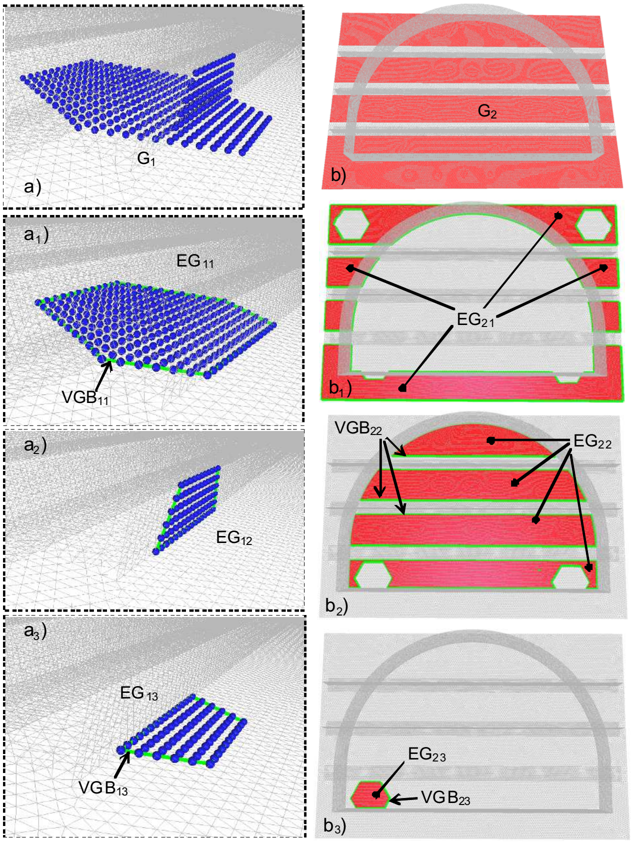


Figure 13. Examples of EGs computed from initial FE groups of the caisson (see figure 12).

REFERENCES

[1] J. Stark. Product Lifecycle Management: Paradigm for 21st Century Product Realisation, Springer-Verlag, 2004.

[2] S. Tichkiewitch, D. Brissaud. Methods and tools for co-operative and integrated design, Kluwer Academic Publishers, 2004.

[3] R. Lou, A. Mikchevitch, J-P. Pernot, P. Véron. Direct merging of triangle meshes preserving simulation semantics for fast modification of numerical models. Tools and Methods for Competitive Engineering (TMCE'08), vol.**1**, pp. 119-131, 2008.

[4] Aim@Shape 2004-08. Advanced and innovative models and tools for the development of semantic-based systems for handling, acquiring, and processing knowledge embedded in multidimensional digital objects. European Network of Excellence, Key Action: 2.3.1.7 Semantic-based knowledge systems, VI Framework, URL: http://www.aimatshape.net.

[5] C.G. Armstrong, D.J. Monaghan, M.A. Price, H. Ou, J. Lamont. Integrating CAE Concepts with CAD Geometry. in Engineering Computational Technology edited by B. H. V. Topping and Z. Bittnar, Saxe-Coburg Publications, pp. 75-104, 2002.

[6] T.De Martino, B.Falcidieno, F. Giannini *A Reference Model for Product Information Sharing in Concurrent Engineering Environment*, Virtual Prototyping: Virtual environments and the product design process, IFIP Chapman & Hall, 1995

[7] R. Bidarra, W.F. Bronsvoort. Semantic feature modeling. Computer-Aided Design, 32(3), pp. 201–225, 2000.

[8] O. Hamri, J-C. Léon, Interoperality between CAD and simulation models for cooperative design. CIRP Design Seminar. May 12-14th, Grenoble, France, 2003.

[9] O. Hamri, J-C. Léon, F. Giannini, B. Falcidieno. Using CAD models and their semantics to prepare FE simulation. ASME DETC Int. Comp. in Eng. Conf., Salt Lake City, 2005.

[10] J-P. Pernot, B. Falcidieno, F. Giannini, J-C. Léon. Hybrid models deformation tool for free-form shapes manipulation. 34th Design Automation Conf. (ASME DETC08-DAC49524), New-York, USA, 2008.