

# AUTOMATIC ATTACHMENT OF BOUNDARY CONDITIONS AND MATERIAL PROPERTIES IN SOLID MODELING PROCESS

Akio MIYOSHI\*<sup>1</sup>

\*1 Department of Quantum Engineering and Systems Science  
The University of Tokyo

E-mail: miyoshi@garlic.q.t.u-tokyo.ac.jp, web page: <http://garlic.q.t.u-tokyo.ac.jp/~miyoshi>

**Key words:** Boundary conditions, Solid models, Material properties, Mesh generation.

## 1 INTRODUCTION

Attachment of boundary conditions, load conditions and material properties to elements and nodes is an important part of generating a mesh although it is rarely discussed comprehensively. If stress analysis is the domain of our interest, boundary conditions are displacement constraints. Assignment of these conditions will be a simple matter to a user if she can assign them directly to a solid geometry model rather than to individual elements and nodes. An element and a node are too fine-grain entities for a user to handle if an ultra-large scale mesh is concerned. The user selects vertices, edges and faces instead of elements and nodes to designate locations of boundary conditions.

The author has been working for developing an agent-supported CAE system on a stand-alone PC [Miyoshi et al. 1999]. Because the CAE system includes a solid modeler as one of the primary components, the Adventure project has determined to use this system as a prototype for the functionality of adding boundary conditions to a mesh data through a geometry model. The scale of models this CAE system can handle is small. However, it is not a problem to use this system as a prototype because boundary conditions are attached to surfaces and points on the surfaces and they can escape difficulties resulting from the ultra-large scale even if this functionality will be implemented on an ultra-large scale. As this system is equipped with an agent to support the user, the agent also handles assignment of boundary conditions to a particular geometrical entity.

Figure 1 presents a classification of boundary conditions and other data attached to the mesh. Shown at the left are element base data such as volume force and material properties and data attached on element faces such as pressure load. Node based data such as point load and displacement constraint is illustrated in the center. At the right is data based on the solid model's feature such as vertex, edge, surface, a part of a surface and a body. Technically speaking, it is not difficult to attach those data to a vertex, an edge, a surface or a body. Therefore this year's work is concentrated on attaching displacement constraints on a part of a surface which requires a strong interaction with the solid modeler's geometry kernel. The solid modeling kernel used in this effort is DESIGNBASE [Ricoh, 1997].

When is it necessary to attach displacement constraint on a part of a surface? Suppose a mechanical part has holes for fasteners. Sometimes it is necessary to have a detailed mesh of to clarify the interaction between a bolt and a hole. But if the bolt and the hole have a large margin of safety, design engineer's interest is placed on other behavior of the mechanical part such as bending strength as a whole. In this case, an analyst would probably need a simpler

model where the part is fastened to another part. As shown at the right of Figure 2, the hole is simply modeled as a circular shape on the fastened surface of the part. The circle is not depressed nor a boss. It is just a two dimensional shape (2D shape for short) drawn on the surface of the part. After meshing, all the nodes on and within the circle will be constrained.

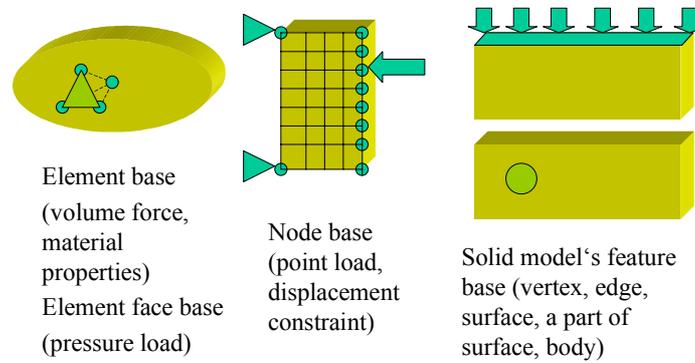


Figure 1 A classification of data attached to a mesh

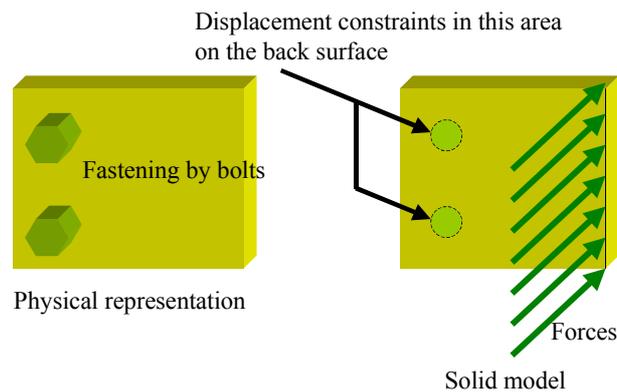


Figure 2 Modeling of bolt fastened holes

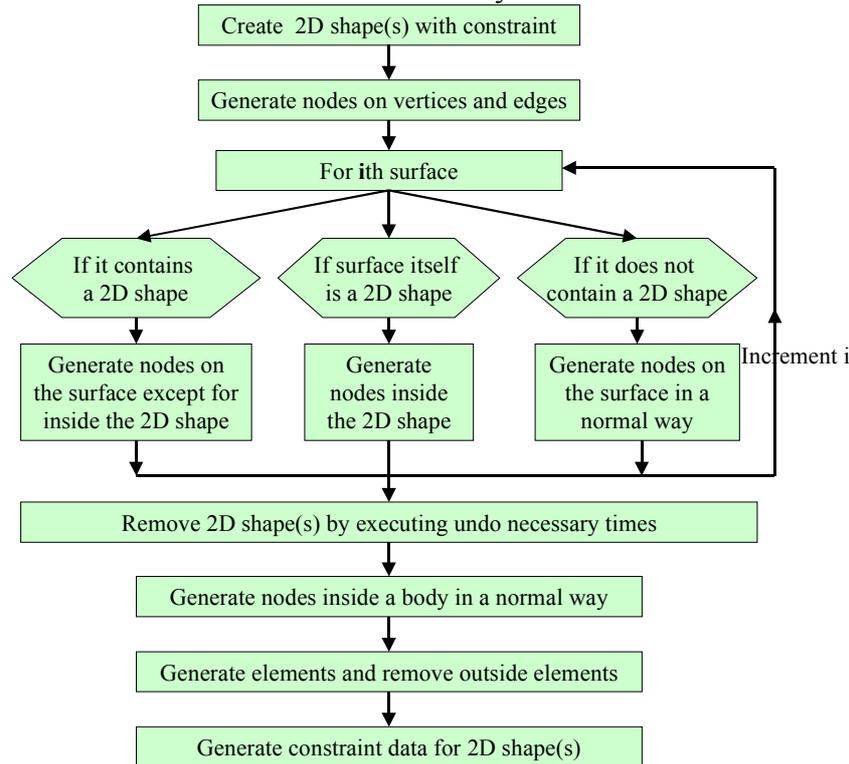
## 2 METHOD

The method of attaching displacement constraints on a part of a surface may depend on how a 2D shape is treated inside a geometry engine. Here, the method for the case of DESIGNBASE is explained.

### 2.1 Analysis flow

The mesh generation approach is the same as that developed by Yoshimura et al. [S. Yoshimura, 1995]. The point of the method to add constraints is how to generate nodes inside a 2D shape and how to avoid 2D shape when generating nodes on a surface that contains 2D shape(s). The node generation flow is illustrated in Figure 3. First, 2D shapes that will be constrained are generated on the surfaces of the solid model. DESIGNBASE calls the surface a loop, i.e., an area surrounded by a loop made of a closed sequence of edges. A 2D shape is created as a loop. Secondly, nodes are generated on vertices and on edges in the same way as

when there is no constrained 2D shape. Then nodes are generated for each surface. There are three branches at this point. If the surface is a 2D shape, nodes are generated inside the 2D shape just as the normal surface. If the surface contains a 2D shape, nodes are generated on the surface avoiding the inside of the 2D shape. At this time, nodes generated must be far enough from nodes inside the 2D shape as specified by the local nodal density. This means that nodes inside the 2D shapes must be generated before any surfaces that contain 2D shapes are processed. The solid modeling kernel's API (application programming interface) is used to decide if a candidate node is inside a 2D shape. If the surface does not contain a 2D shape, nodes are generated on the surface in the normal way.



**Figure 3 Flow of node generation**

After the iteration over all the surfaces is finished, all the 2D shapes will be removed from the solid model. According to the internal representation of DESIGNBASE, if a surface is created on another surface, it must be a hole. Otherwise the solid model has self-contradiction and no more of its manipulation is possible. As the 2D shapes were created as surfaces on a surface without making a topology of a hole, the solid model has a self-contradiction. Only reference to geometrical information is possible at this point. So, all the 2D shapes must be removed immediately after all the nodes are generated on surfaces and before any geometrical manipulation is performed during node generation inside the body.

After elements are generated and elements outside the solid model domain are removed, displacement constraint data is generated. At this point, it is no more necessary to distinguish nodes on 2D shapes and those on regular surfaces.

### 3 RESULTS AND DISCUSSIONS

Figure 4 presents a process of node generation on 2D surfaces. A cube is selected as an example solid model. A surface of the cube has two circles. The nodes on the circle at the upper left is to be constrained while those on the lower right circle will not have any constraint. The figure at the left shows how nodes are generated on a 2D shape in red. Nodes on a circular boundary have already been created as shown in deep blue. Then, nodes are generated inside the second 2D surface. Lastly, nodes are generated on the square surface that contains the two circles. Note that they avoid circles and the distance between nodes across the circular boundary is kept greater than a certain value.

Figure 5 shows a Von Mises equivalent stress contour plot obtained from this mesh when a load indicated by an arrow is applied. Shape before deformation is shown in a thin violet line. A steep stress gradient can be seen across a circular boundary showing that the cube was constrained here. Automatic attachment of boundary conditions to a part of a surface was successfully demonstrated.

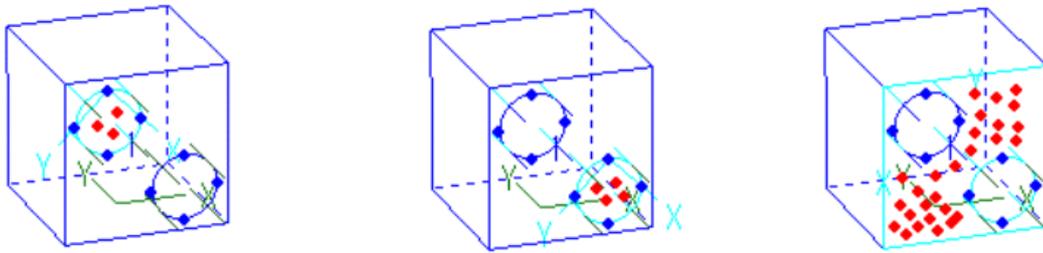


Figure 4 Process of node generation on 2D shapes and on a surface containing 2D shapes

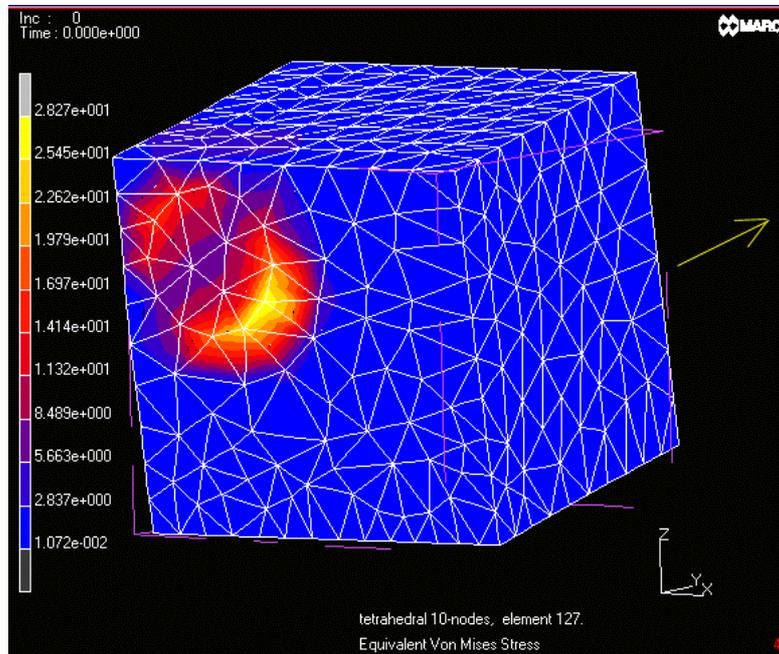


Figure 5 An example mesh with a part of a surface constrained

## 12 CONCLUSIONS

A method to attach displacement constraint on a part of a surface through a solid model is proposed and implemented to an agent-supported CAE system on a stand-alone PC. An example successfully demonstrated the validity of this method. Since this type of boundary condition data is for surfaces only, it is not necessary to consider a parallel processing. Although the method was implemented only for a planar surface, it can be readily applied to surfaces with curvature.

## REFERENCES

- Miyoshi, A., Yagawa, G. and Sasaki, S. (1999) An Interface Agent that Actively Supports CAE Beginner Users in Performing Analyses, *Advances in Engineering Software*, Vol. 30, No. 8, pp. 575-579
- Ricoh company, Ltd.(1997), *DESIGNBASE User's Guide V5.2*
- Yoshimura, S., Lee, J. S. and Yagawa G. (1995) Automated Analysis of Stress Intensity Factor for 3-D Cracks, *Proc. 5<sup>th</sup> JSME/ASME Joint International Conference on Nuclear Engineering*, Kyoto, Japan, 1, pp. 357-362