Meshing of Tolerant Models in the Finite Octree Mesh Generator

Marcel K. Georges
Scientific Computation Research Center
Rensselaer Polytechnic Institute

January 12, 1994

Recent improvements in geometric modeling practices allowing the assignment of different geometric tolerances to different model entities improve the robustness of the modeling environment and allow the definition of clean models. A clean geometric model does not contain extremely small geometric feature(s) relative to the size of the geometric model. Geometric modelers using a single geometric tolerance may create undesirable small geometric entities due to model definition input which is not as accurate as the model tolerance. Although model input data may be accurate to a modeler defined precision, small geometric entities may be created due to round off in geometric computations. The existence of small geometric entities in a geometric model may not be desirable for some applications such as automatic mesh generators employing no knowledge of the shape and topology of geometric models.

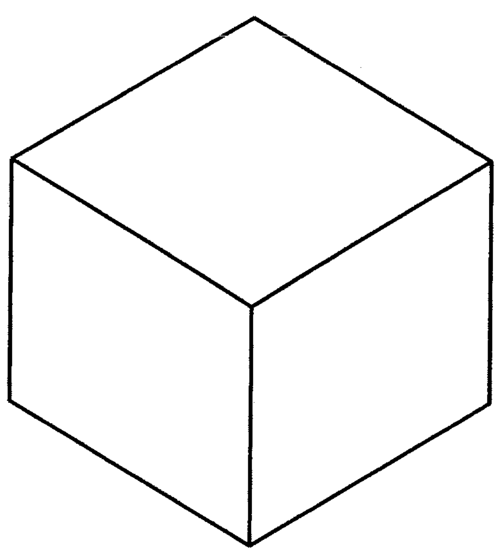
The Finite Octree interrogates the geometric modeler supporting the geometric model definition to generate a mesh. To ensure validity of a generated mesh, a set of requirements such as "topological consistency" and "geometric similarity" has to be maintained. To ensure these requirements, ill shaped finite element entities classified on small geometric features are created. To generate meshes with acceptable shaped elements that can be used to perform a finite element analysis requires a lot of refinement in the neighborhood of small geometric features, resulting in the generation of meshes with a very large number of elements.

Tolerant modeling improves the robustness and computational efficiency of a modeling environment. It allows accurate modeling computation for models defined in a different modeler using different tolerance schemes. For example, importing a very simple model defined in a modeler using a tolerance of 10^{-6} results in the definition of an invalid model with gaps near the boundary of model faces. Although tolerant modeling practice eliminates extremely small geometric features from the definition of a geometric model, it may not eliminate relatively small features. The issue of generating finite element meshes without ill shaped elements can be achieved by violating the mesh validity checks.

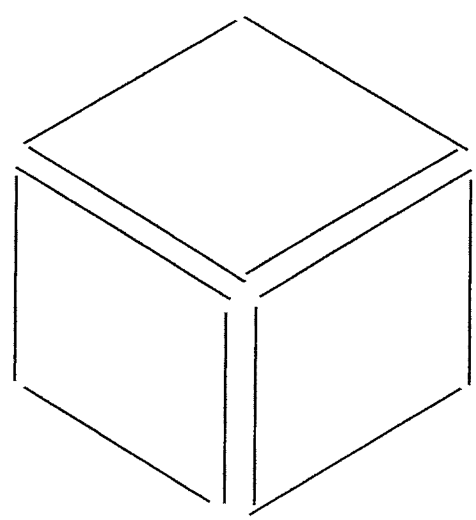
Before discussing the concept of tolerant modeling, an overview of how geometric models are defined within geometric modelers is provided. This will lead to a better understanding of the concept and shows that some of the geometric modelers interfaced with Finite octree have been using this concept and shows the importance of modification to the tolerancing algorithms used by the Finite Octree to improve the robustness of the meshing process.

Geometric model definition in geometric modelers interfaced with Finite Octree.

The geometric modelers that have been interfaced with Finite Octree deal with model definitions in a consistent manner, when a model is viewed as a set of topological entities and geometric maps defining the shapes of the model entities. For two-manifold models, model faces are defined typologically by a set of loops consisting of all model edges bounding the face. Geometrically the face is defined by the mathematical definition of the map and a set of curves on the map defining the bounds of the face. A model edge is topologically represented by one or two model vertices. Geometrically it may or may not have a single geometric definition, map, defined in R^3. However, each model edge is represented by a curve defined on the map of each model face bounded by the edge. Each curve on any map has begin and end points defined on the map of the face. End and begin points of two consecutive curves are guaranteed to be within a given tolerance. Figure 1 shows how a simple cube is modeled. Figure 1a shows the topological representation of the model. Figure 1b shows the geometric definition of all curves, defined on maps of model faces, that are commonly defined in all geometric modelers. (Note: geometric representation of vertices is not shown because their definition is not supported by all geometric modeler).



a)



b)

Figure 1: a) Topological representation of a cube; b) Common geometric representation of topological entities between all modelers.

Different modelers may deal with this model in a different manner. For example a geometric modeler may not provide a single geometric definition for each model edge or vertex. However, all modelers assure that the defined curves for a model edge are within a certain tolerance. Similarly, the end and begin points of two consecutive curves for each face are also within a certain tolerance. However, having two points of two consecutive curves does not yield that all points of all curves meeting at a vertex are within that tolerance. This is demonstrated by the simple 3–point distance problem. In figure 2, the distance between 1–2 and 2–3 is within a certain tolerance. However, the distance 1–3 is greater than the tolerance.

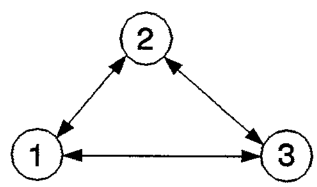


Figure 2: Three point tolerance problem.

In Parasolid, when the modeling environment tolerance is used, a single spatial location is created for each vertex, and a curve in R^3 is created for each edge. It is assured that a generated curve for a model edge in R^3 is within the modeling environment tolerance of all curves on the surfaces defining model faces bounded by the edge. The begin and end locations of the curve in R^3 are also within the modeling environment tolerance from the beginning and ending vertices of edges. However, when a different tolerance is assigned to an edge, the curve in R^3 defining the edge is deleted and replaced by two different curves on the surfaces of the faces bounded by the edge. A spatial location is then assigned to bounding vertices such that all corresponding begin or end points of curves on the surfaces are within the edge tolerance from vertex location. The generated curves of the edge on each surface are within the assigned tolerance. Figure 3a shows the geometric representation of the cube with modeling environment tolerance assigned to all model edges. Figure 3b shows the geometric representation when a tolerance different than modeling environment is assigned to one of the edges. The bolded lines represent the geometric definition of model edges assigned the modeling environment tolerance. The non-bolded lines represent the curves on the maps of surfaces. The points represent the location of model vertices. The curves on surfaces for non

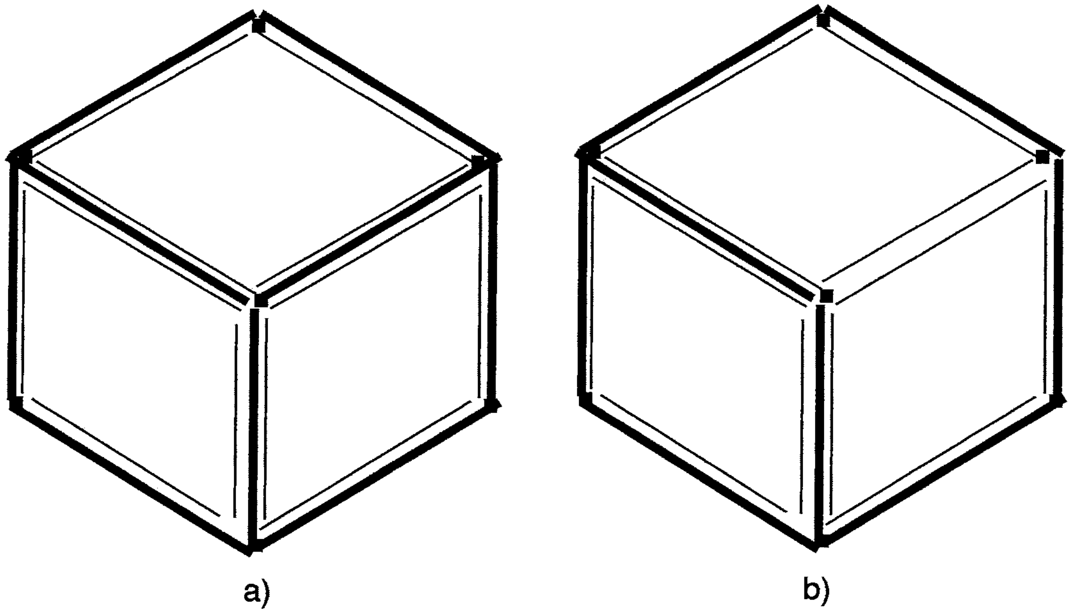


Figure 3: Geometric definition of cube within Parasolid. a) Modeling environment tolerance assigned to all edges;
b) A tolerance is assigned to one model edge.

tolerant edges are not available for direct interrogation. Curves on surfaces for tolerant edges are available and can be interrogated to get pointwise geometric data.

CATIA does not support model vertex definition, but a single geometric definition is defined for each edge. The curves on the faces are also available. The geometric definition of the model is shown in figure 4. It seems that the end points of edges meeting at a vertex are not always within a known tolerance. This is also similar to the three point tolerance problem. To deal with this problem effectively a location for any model vertex must be computed and an appropriate tolerance must be assigned to the vertex. The tolerance is the radius of the smallest sphere defined at the computed location that contains all end points of edges. All intersections performed to find vertices must be eliminated and topological queries must be used to find all points and curves that must be considered to find the location of a vertex. A separate document describing how to improve the robustness of the integration of Finite Octree with CATIA will be provided.

Acis and Shapes are similar to Parasolid when no tolerance other than modeling environment tolerance is assigned to a model entity. The geometric representation will be similar to what is shown in figure 3a. CATIA is slightly different than the other three modelers. Because ending points of edges are not guaranteed to be within a specified tolerance, a problem may exist in the creation of vertices.

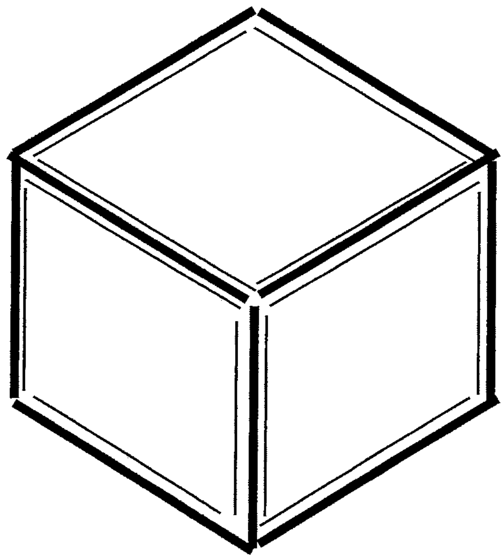


Figure 4: Geometric definition of cube within CATIA.

The difference between the two representations can be resolved when model entities are assumed to touch at their boundaries within different tolerances. Currently, geometric models with different modeling tolerances assigned to various topological entities can not be meshed by Finite Octree. Algorithmic procedures are being changed to allow meshing of such models ("tolerant model").

Definition of a tolerant model

This section is intended to define what constitutes a valid "tolerant model". In a tolerant model, each topological model entity $({}_GT_i^d, d = 0, 1, 2)$ is assumed to have a tolerance value $({}_tT_i^d, d = 0, 1, 2)$ and a tolerance region whose shape depends on the dimension, d, of the topological model entity. A topological model vertex ${}_GT_i^0$ is represented as point X in R^3 with a tolerance defined as a sphere centered at X with a radius equal to ${}_tT_i^0$, figure 5a. A model edge ${}_GT_i^1$ is represented as a 1–D locus of points in R^3 with a tolerance region defined as the 1–D locus of spheres centered along the 1–D locus of points with a radius equal to ${}_tT_i^1$, figure 5b. A model face ${}_GT_i^2$ is represented as a 2–D locus of points in R^3 with a tolerance region defined as the 2–D locus of spheres centered along the 2–D locus of points with a radius equal to ${}_tT_i^2$, figure 5c.

A list of important notes is provided here to qualify some of the properties of tolerant models.

1. The location of a model vertex is within the assigned tolerance to a point on a higher order entity bounded by the vertex. However, the location of the vertex may not

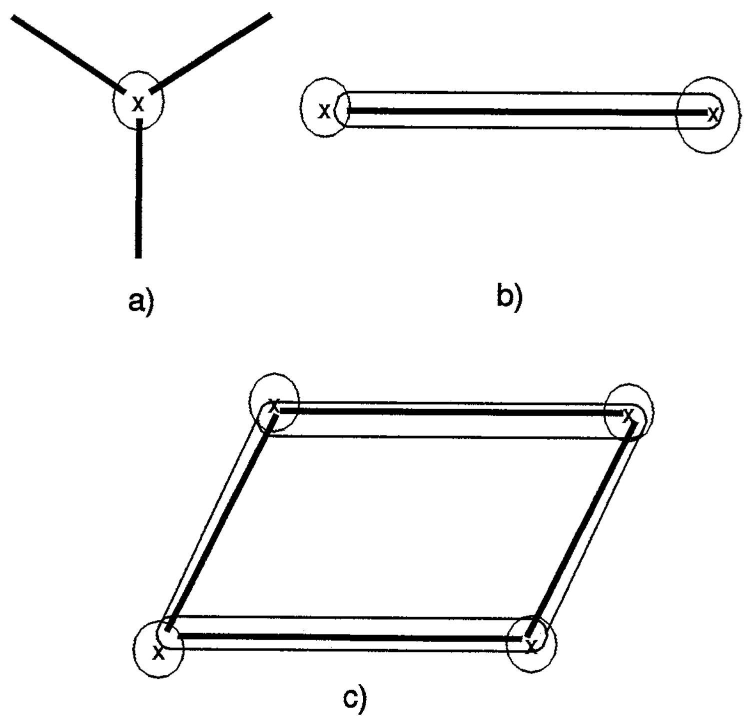


Figure 5: Geometric representation and tolerance region of a model vertex, edge, and face.

be within the tolerance of a higher order entity bounded by the vertex. This can be true only when the vertex is assigned a larger tolerance then the higher order entity bounded by the vertex.

2. Any spatial point provided by the modeler as on a model edge is within the assigned tolerance to the curve on the surface defining a bounded face by the model edge.
3. A point on the surface of a face that is within the tolerance region of a model edge bounding a face defined on the surface may be up to twice the edge tolerance away from another surface defining another face bounded by the same edge.

The above list indicates that if a spatial location is classified as within the tolerance of a model vertex , it can not be assumed to be within the same tolerance from a model edge or face bounded by the vertex. The same argument is true for points classified on model edges.

When the Finite Octree algorithms operate under the above listed properties, the differences between tolerancing schemes used by different modelers can be eliminated by selecting the proper tolerances for model entities.

Definition of a tolerant model to be meshed by Finite Octree.

A valid geometric model that can be meshed by Finite Octree must satisfy the following requirements:

1. The tolerance of a model entity is greater or equal to the tolerance of all higher order entities bounded by the entity, $\left\{ \left({}_tT_i^d \geq {}_tT_j^{d+1} \right) \mid {}_GT_i^d \in \left\{ \partial \left({}_GT_j^{d+1} \right) \right\}, d = 0,1 \right\}$. (i.e. Tolerance of a model edge is greater or equal to the tolerance of all model faces bounded by the edge. Tolerance of a model vertex is greater or equal to the tolerance of all model edges and faces bounded by the vertex.)
2. The tolerance regions of two model entities not sharing common boundaries can not come in contact. This condition indicates the following:
    a. Model vertices can not approach within the sum of their tolerances.
    b. Any model edge must have a length greater than the sum of the tolerance of the two bounding vertices.
    c. The closest distance between two model edges not sharing a model vertex must be greater than the sum of the two edge tolerances.
    d. The closest distance between two model faces not sharing a model edge or vertex must be greater than the sum of the two face tolerances.
    e. The closest distance between a model edge and a model face not sharing a model vertex must be greater than the sum of the edge and face tolerances.
    f. The closest distance between a model vertex and a model edge or face not bounded by the vertex must be greater than the sum of the tolerances of the two entities.
3. The tolerance regions of two model entities sharing common boundaries must separate and can not come in contact after separation. This indicates the following:
    a. No two model entities are completely in contact.
    b. Two model edges sharing a common vertex can lie within the sum of the edge tolerances in the neighborhood of the vertex. They can not come in contact after separation.
    c. Two model faces sharing a common vertex or edge can lie within the sum of the face tolerances in the neighborhood of the common entity. They can not come in contact after separation.
    d. A model edge or a face can not self intersect. Two distinct points on a model edge can be close to within twice the entity tolerance when any point on the edge between the two points is within the edge tolerance from the line between the two points. The edge in figure 6a shows a non self intersecting edge. Figure

6b shows a self intersecting edge, because the closest distance between the third point on the curve and the straight line between the two points within twice the edge tolerance is greater than the edge tolerance. This restriction leads to limiting the smallest radius of curvature along a model edge or face to the assigned tolerance.

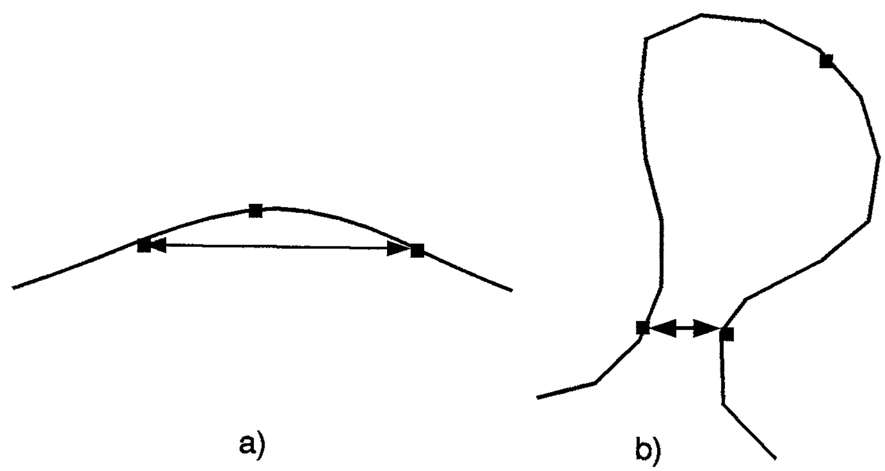


Figure 6: Self and non self intersecting edges.

All available commercial geometric modelers claim to satisfy the above requirements. The condition and type of interaction allowed between model entities in the neighborhood of a common lower order entity is not discussed. Currently, Finite Octree stops when two model entities come in contact within the modeler tolerance. The initial study indicates that when two model entities are within the sum of their tolerances, interpenetration between these entities may be allowed, figure 7a. In addition, pointwise geometric interrogations may not always yield a sufficient or consistent answer. Geometric operations requiring more than pointwise geometric interrogation may not provide the needed data. When a plane orthogonal to the planar face, shown in figure7a, is intersected with the face, most modelers will split the face into three faces as in figure 7b. The portion of the two edges within the tolerance is merged into one edge and one new model vertex is created.

The problem of close geometric entities within the sum of their tolerance is not addressed in this document. It will be addressed in the near future. The immediate changes to the Finite Octree algorithms will deal with the type of models that currently can be meshed with the extension of different modeler tolerances for various model

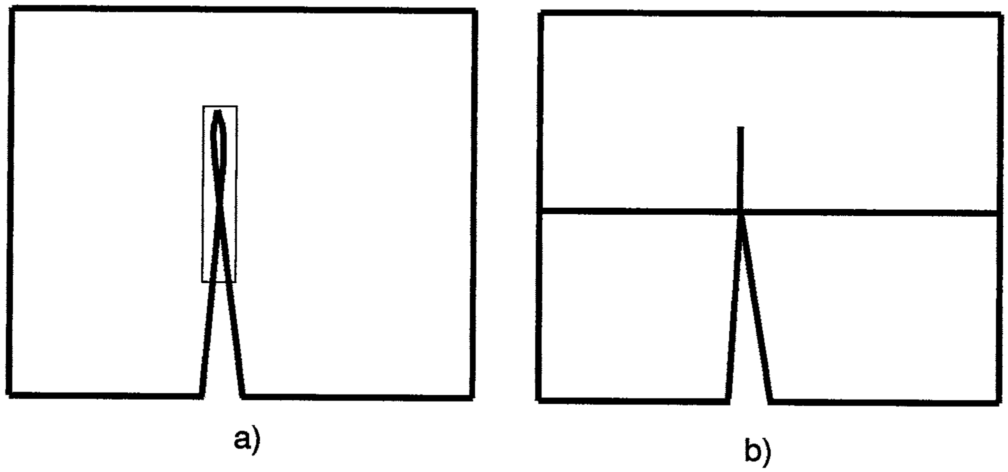


Figure 7: Intersecting two edges sharing a common bounding vertex.

entities. These changes involve the changes of point classification with respect to the octree. Any mesh vertex classified on the boundary of the geometric model will be classified with respect to the octree using the modeler tolerance assigned to the model entity. This may involve some other changes such as classifications of mesh edges and faces with respect to octants. Any classification of mesh edges or faces depending on pointwise classification may result in a missclassification and a failure in the meshing process.

With respect to Parasolid, when an edge has a tolerance assigned to it, one of the curves on one of the surfaces will be used as the geometric definition of the edge in R^3. This decision has been taken after discussion with Shape Data. By taking the curve on the surface of the first face known to the edge as the geometric definition of the edge, the other curve on the second surface will be within the tolerance region of the edge. This indicates that the maximum distance of any point on one curve from the second curve is less than the tolerance value of the edge. Minor changes to the parasolid interface are needed to deal with the KI changes in Parasolid.

Besides the changes to deal with point classifications with respect to octants, most of the changes to deal with this problem are hidden in the interfaces to the geometric modelers. With respect to Finite Octree the model is defined as given in the first section. The difference in approaches needed to resolve the difference between different modeling systems is hidden inside the interfaces. With the proposed tolerancing approach all geometric modelers defined in the four geometric modelers mentioned above can be meshed with proper changes to their interfaces.