***** WORKING DRAFT *****

Specification of Mesh Control Functions in Finite Octree

SCOREC Report # 5-1994

**Rao Garimella, Saikat Dey,
Ravichandran Ramamoorthy,
Marcel K. Georges and
Mark S. Shephard**

Scientific Computation Research Center,
Rensselaer Polytechnic Institute,
Troy, NY 12180.

March 1, 1995

Contents

1 Introduction

Finite Octree is a fully automatic mesh generator that can mesh general non-manifold solids[1] [3]. Currently, Finite Octree supports the generation of tetrahedral and hexahedral solid, triangular surface, and line elements.

Finite Octree interfaces directly with geometric modeling systems through a fixed set of operators which answer geometric and topological inquiries by the mesh generator. Finite Octree is available with interfaces to Parasolid™, Catia™, Shapes™, ACIS™ and SCOREC 3D geometric modelers.

This document describes the functions for *a priori* control of mesh gradations and element sizes in Finite Octree meshes [3].

1 Non-manifold objects are general combinations of wires, surfaces and solids and may have surfaces touching at an edge or a single point.

2 Exercising mesh control

The accuracy of a finite element solution depends on the distribution of elements throughout the domain. For this purpose Finite Octree supports a wide range of methods to control element sizes and mesh gradations. Some of these controls place explicit bounds on element sizes while others control element shapes and geometric approximation errors.

Since users of Finite Octree or applications interfacing to it only have knowledge of the geometric model but not about the mesh, mesh control parameters or *meshing attributes* must be assigned to topological entities of the geometric model. Some of the mesh controls may also be applied to spatial points[2]. Meshing attributes may be applied to the entire model (global mesh control) and/or to individual topological entities (local mesh control). Finite Octree obtains the meshing attributes on a model entity through operator so that the actual mechanism of associating the meshing attributes with model entities is hidden from the mesh generation procedures. Thus, applications interfacing to the mesh generator can use Finite Octree's meshing attribute retrieval operator or replace it with one specific to the application.

The next three sections discuss the types of mesh control currently available in Finite Octree, the mechanism used by Finite Octree to associate meshing attributes with the model being meshed and the structure of the meshing attribute retrieval operator.

2.1 Understanding mesh controls

The following are the different types of mesh control that can be exercised in Finite Octree

1. Element sizes
2. Number of elements
3. Curvature dependent refinement, and
4. Detection of small model features and control of element aspect ratios (longest edge to shortest edge ratio) in their vicinity
5. Elimination of small mesh entities caused by small model entities

[2] Currently, this type of mesh control is available only to applications interfacing with Finite Octree procedures

Generation of elements of desired sizes can be controlled in Finite Octree by explicitly placing upper and lower limits on the sizes. Element sizes are specified as a linear dimension smaller than or equal to the model size. The model size is defined as the longest dimension of the geometric model along the coordinate directions.

The maximum element size controls how fine the discretization of an entity is going to be while the minimum element size places an upper limit on how much refinement can be done by any other mesh control function. *Note that the upper limit on element sizes may be violated by Finite Octree to ensure that any portion of the mesh is topologically equivalent to the original model in that region.*

Element size controls can be used very effectively to obtain different mesh densities in different regions of the model. *However, it is important to note that Finite Octree can only satisfy the element size requirement approximately.* The effect of specifying different values for maximum element size is illustrated in Figure 1. Figure 1a shows the geometric model to be meshed. The size of this model is 1.828 units. The Finite Octree mesh with a requested element size 0.30 units is shown in Figure 1b while the mesh with a requested element size of 0.15 units is shown in Figure 1c.

It is also important to note that in the event that different maximum element size limits are placed on a model entity and higher order entities adjacent to it, the minimum of these upper limits on element sizes is adopted for the entity. Thus, if the maximum element size for a model face has been given a value larger than the regions it is connected to, the value for the face is overridden.

Another method of regulating element sizes is controlling the number of elements on an entity. Finite Octree currently offers this type of control only for the entire model, not for individual entities. Thus, upper and lower bounds can be placed only on number of elements generated across the longest dimension of the model. This type of control too will generate only approximately the requested number of elements.

Curvature-based refinement is used to automatically improve the approximation of the model geometry by the mesh in the vicinity of curved boundaries [1]. Thus the curvature based refinement procedures can automatically create meshes that are finer in regions of high curvature and coarser in regions of low curvature. These procedures control mesh refinement and gradation by regulating the geometric approximation error below user specified limits. The geometric ap-

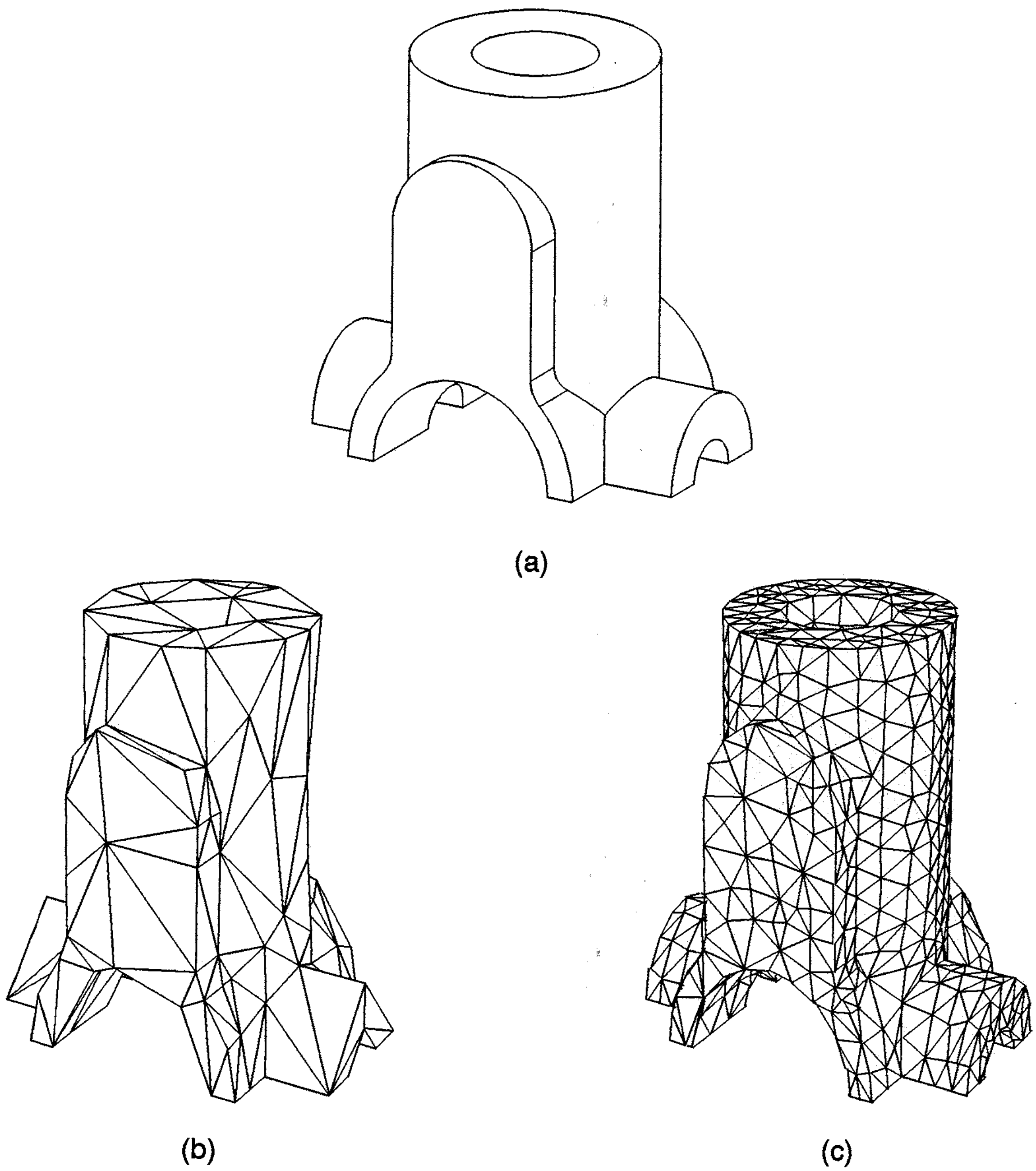


Figure 1 Illustration of the effect of controlling the maximum element size a) geometric model of size 1.828 units b) mesh with maximum element size of 0.30 c) mesh with maximum element size of 0.15

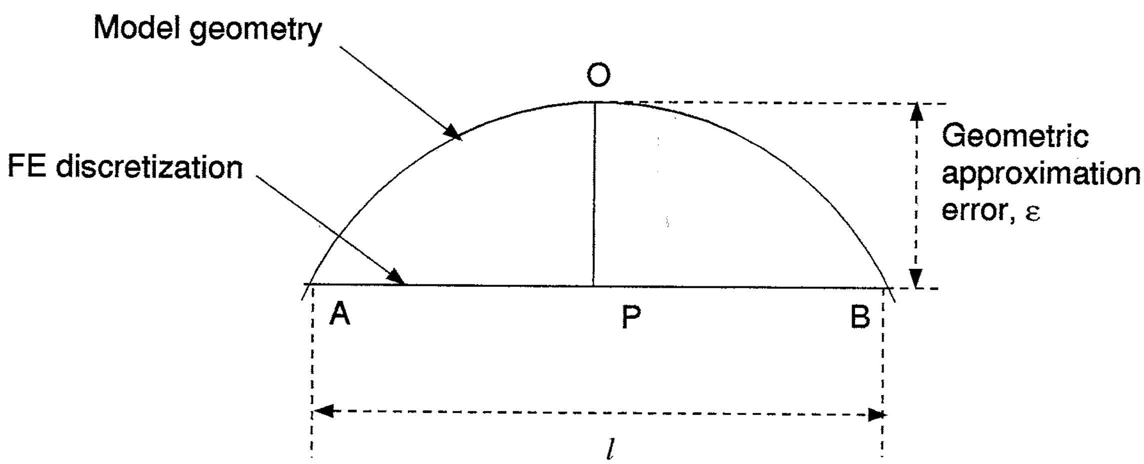


Figure 2 Illustration of definition of geometric approximation

proximation error, ϵ, is defined as the maximum distance between a finite element entity and the portion of the model geometry which it approximates. Figure 2 illustrates the definition of geometric approximation error for a model edge and its finite element discretization.

The refinement and gradation produced in the mesh depends on the method of specifying the geometric approximation error. Finite Octree supports two methods for specification of geometric approximation error by the user. The first method maintains a constant geometric approximation error for mesh entities classified[3] on a particular model entity. The constant value of the geometric approximation error is specified either as an explicit value or as a fraction of the model size. The second method specifies a variable geometric approximation error for mesh entities classified on a model entity. The geometric approximation error is specified as a fraction of the mesh size represented by the finite element edge length.

For constant curvature surfaces (e.g. cylinders, spheres), either method will result in uniform refinement over the face. On the other hand, the second method tends to be better suited for surfaces with large curvature variations (e.g. parametrically defined spline surfaces, Bezier surfaces). This is because attempting to maintain the geometric approximation error below a constant value over the entire entity by the first method typically generates more elements for meshes with the same size smallest element. However, the second method of curvature-based

[3] The unique association of a topological mesh entity with a model topological entity of equal or higher dimension is called *classification[3]*

refinement can predict very small element sizes near curvature singularities and hence this method must be used with a suitable limit on the minimum element size (Reference [1] details the relationship between the geometric approximation error, the curvature of a surface and finite element edge lengths).

Figure 3 illustrates the difference between the two methods of curvature dependent refinement for a model whose top face has a highly curved parametric surface. Figure 3a shows a geometric model with the top face of variable curvature, figure 3b shows the mesh without curvature-based refinement, and figures 3a 3b show meshes generated with the first and second methods of curvature-based refinement respectively. The curvature refinement parameters in the example have been chosen such that the element sizes are approximately equal in the vicinity of the highest curvature. From the figure it can be seen that the second method of curvature-dependent refinement captures the curved feature of the model as well as the first method but does so with fewer elements.

Like the element sizes, the geometric approximation error is maintained below the user specified limit only approximately by the curvature-based refinement procedures.

Another type of mesh control supported in Finite Octree is the detection of model features that are very small compared to the model size and control of element aspect ratios[4] in the vicinity of the feature. In the absence of this control, elements with unacceptable shape tend to be generated in the vicinity of the feature which is much smaller than the requested element size. This control is specified by the user as the maximum allowable ratio of the longest edge to the shortest edge for elements which have at least an edge classified on the feature. As with other mesh controls, this requirement on the aspect ratio of the element is satisfied only approximately.

Figure 4 illustrates the effect of applying small feature checking to a model. Figure 4a shows the geometric model to be meshed and Figures 4b and 4c show the mesh generated without and with small feature checking respectively. The accompanying table shows the improvement in some of the element characteristics of the mesh when small feature checking is enabled.

Users can choose to eliminate small geometric model features from the final mesh. The elimination of small mesh features created as result of the presence of small model features is controlled by two user prescribed parameters.

4 The aspect ratio used here is the ratio of its longest edge to its shortest edge since it is inexpensive to calculate.

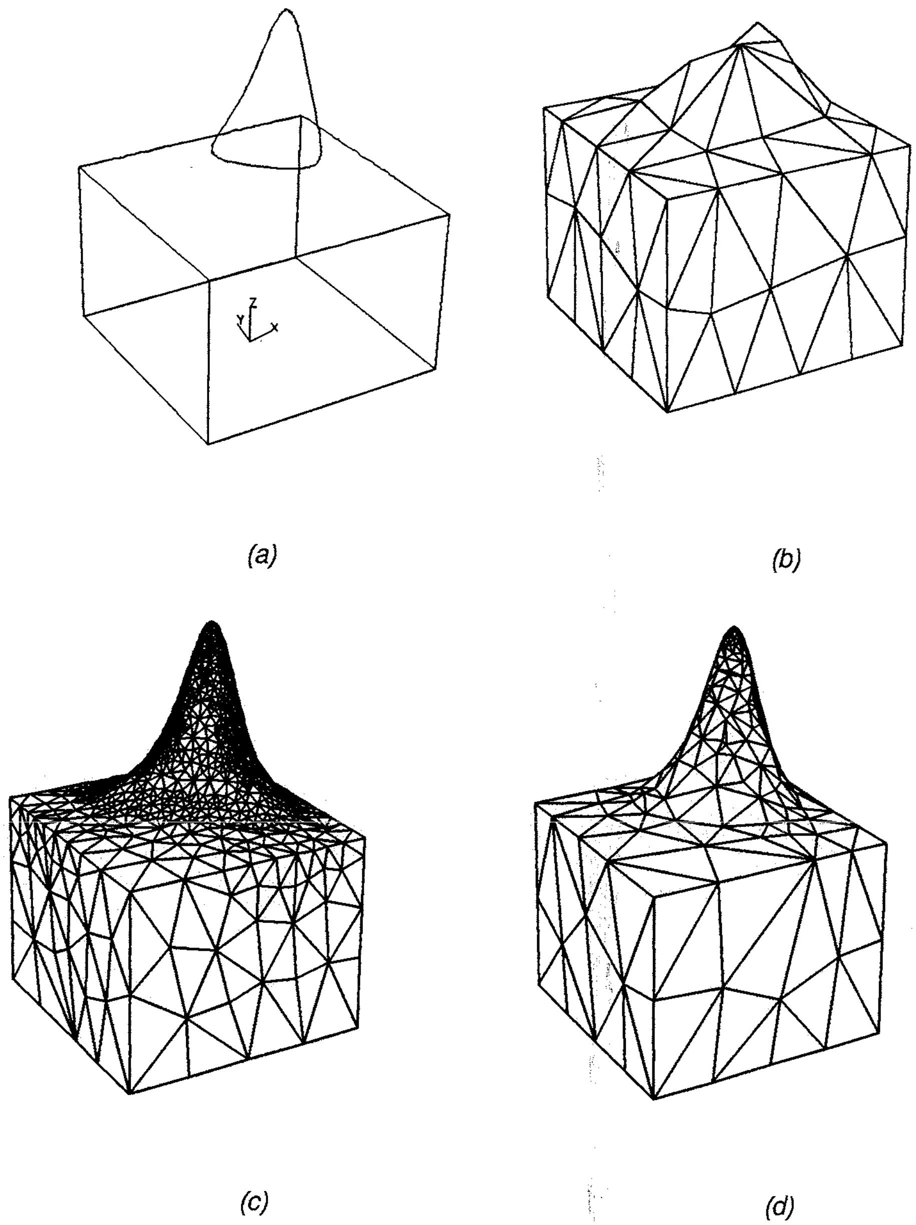
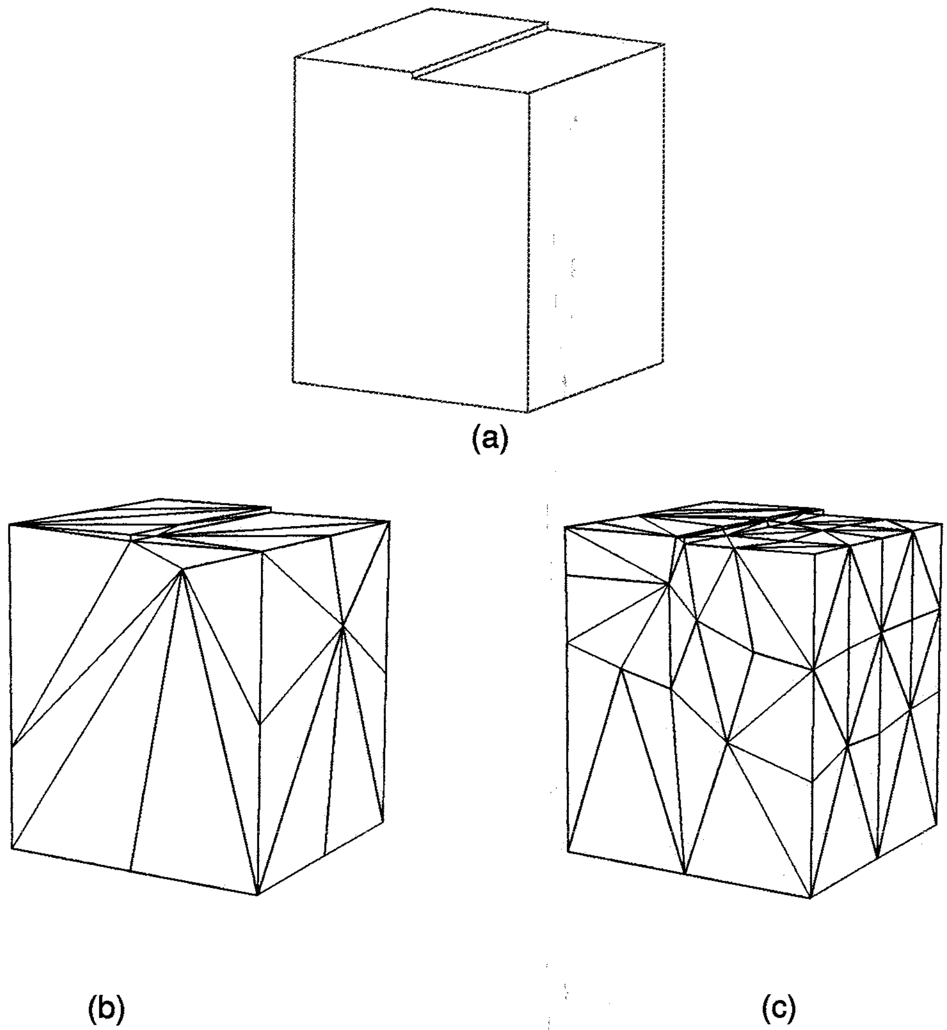


Figure 3 Illustration of two methods of curvature-based refinement a) Geometric model of block with highly curved top face b) Mesh without curvature-based refinement c) Mesh with first method of curvature-based refinement d) Mesh with second method of curvature-based refinement



Element Characteristic	*Small Segment Checking*	
	Off	***On***
Longest edge to shortest edge ratio	*34.143*	*14.431*
Longest edge to shortest height ratio	*46.777*	*21.875*
Largest dihedral angle	*173.978°*	*164.511°*

Figure 4 Illustration of the effect of small segment checking a) Geometric model with small feature b) Mesh without small feature checking c) Mesh with small feature checking (user specified upper limit of element aspect ratio = 25)

1. The lower limit on the size of the model feature to be retained in the final mesh.
2. The upper limit on the aspect ratio of the elements near small model feature.

Any mesh edge that is smaller than the specified lower limit in length and is connected to a mesh face with aspect ratio greater than the specified upper limit will be eliminated from the mesh. Note that the second parameter is required to be able to distinguish between small mesh edges created as a result of small geometric model features and those created as a result of user prescribed mesh refinement. In the absence of the second parameter the procedure will remove all user prescribed mesh refinement which is not warranted. The second parameter is also the aspect ratio limit used to do refinement to improve aspect ratios in the vicinity of small model features as mentioned previously. Like other mesh control options the option to eliminate small mesh entities created by the presence of small model entities can be exercised globally for the entire model or on individual model entities. The global specification is useful when the user does not know the exact details of the model features which are small.

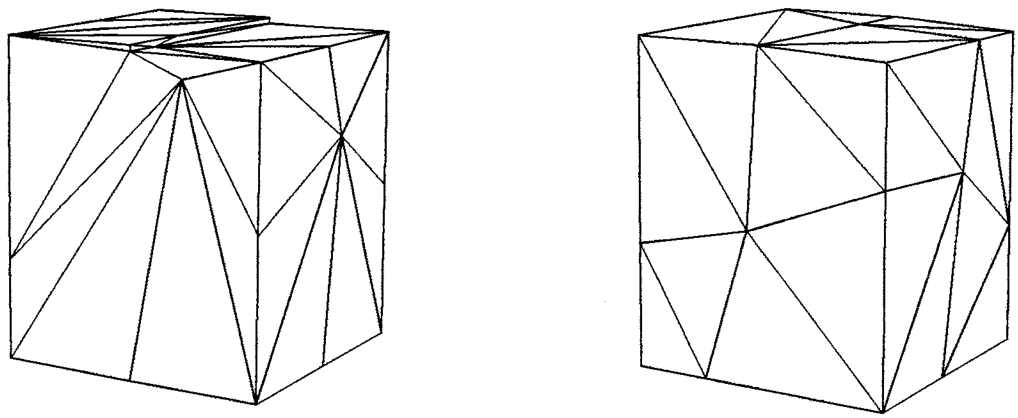
Figure 5 illustrates the effect of applying small mesh entity elimination to a model. Figures 5a and 5b show the mesh generated without and with small mesh entity elimination respectively. The accompanying table shows the improvement in mesh statistics.

The use of the option to eliminate small mesh entities has shown considerable improvement in the element quality and also resulted in significant saving in the total meshing times owing to reduced timings for element shape improvement procedures.

2.2 Assigning meshing attributes in Finite Octree

This section describes the method for assigning meshing attributes if the mesh attribute retrieval operators of Finite Octree are used (See Section 2.3 for a description of how to replace the current mesh attribute retrieval operator with an equivalent operator).

Mesh controls in Finite Octree can be assigned to the entire model (global mesh control) or to individual topological entities of the geometric model (local mesh control). Meshing attributes of an individual entity, when defined, override the global attribute. All of these meshing attributes are read from a meshing



(a) (b)

Element Characteristic	Small Mesh Entity Elimination	
	Off	On
Longest edge to shortest edge ratio	34.143	3.174
Longest edge to shortest height ratio	46.777	3.641
Largest dihedral angle	173.978°	131.439°

Figure 5 Illustration of the effect of small mesh entity elimination a) Mesh without small mesh entity elimination b) Mesh with small mesh entity elimination (user specified lower limit of entity size = 0.1, upper limit of element aspect ratio = 25)

attributes file called ***modelname*.atb** where ***modelname*** is the name of the geometric model being meshed. The first part of the meshing attributes file consists of global meshing attributes for the entire model. In the absence of a meshing attribute on any model entity, the global value of that meshing attribute controls mesh gradations and element sizes on all the entities. One or more of the global values of meshing attributes on a model entity can be overridden by assigning them a different value for that entity in the meshing attributes file. Also, if a particular model entity is assigned different values of the same mesh control, only the last specified value of that mesh control parameter is used. Local mesh control specification follows the global mesh control specification in the meshing attributes file.

Figure 6 shows the format of the meshing attributes file.

AUG	*USX*	*USY*	*USZ*						
ESM	*MNS / MXN*	*MXS / MNN*	*CRM*	*GAT*	*GAP*	*SSF*	*SSR*	*SFF*	*SIZ*
NLR									
ENT	*ENN*	*MAT*	*Variable number of records depending on MAT*						

.
.
.

Figure 6 Format of the meshing attributes file

The first field of the first line is a flag indicating whether periodic surfaces should be augmented or not. In order to efficiently produce a valid mesh, it is of interest to make use of the parametric spaces for surfaces provided by the geometric modeler. However, parametric spaces may not have a 1-to-1 correspondence with periodic surfaces. By splitting a periodic surface, this 1-to-1 correspondence is artificially achieved (Figure 7). The original model topology is retrieved at the end of tetrahedronization and mesh entities are reclassified accordingly[4]. It has been seen that augmentation makes the meshing process more efficient for models with periodic surfaces. Note that not utilizing this feature will still produce valid meshes; however more refinement may be required to resolve topological incompatibilities. (Default — Off)

The next three numbers are scaling factors (in the three coordinate directions) for the universe Finite Octree uses for meshing. The universe, as defined in Finite Octree, is the bounding box of the model, magnified by 10% and scaled in the appropriate coordinate directions by the three scaling factors. The default value for each of the scale factors is 1. Therefore, by default, Finite Octree uses a cubical universe to mesh the model. Say, the X scaling factor (field *USX*) is set to 10 and the others are allowed to remain at the defaults, the meshes obtained would then be stretched in the X direction. The three scaling factors may also be lesser than 1 in any direction, but care must taken that the resulting universe size is not smaller than the model extents in that direction.

The second line of the file contains global values of mesh control, the third line specifies the number of local mesh control records that follow it and the rest of the file contains the local mesh control information for individual model

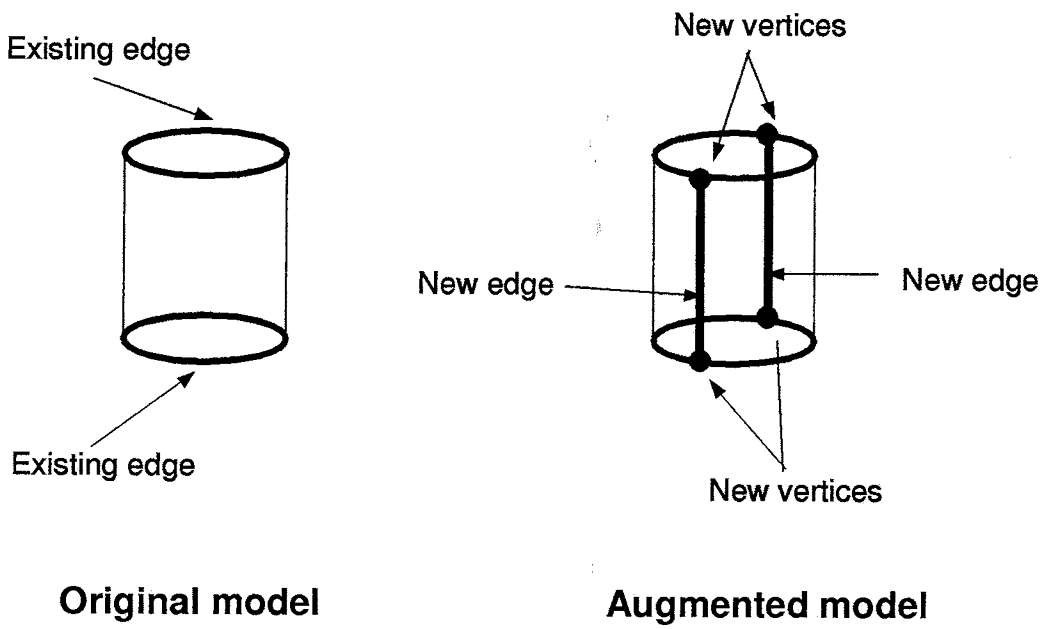


Figure 7 Augmentation of models with periodic surfaces

entities. An explanation of the global mesh control parameters is given below followed by an explanation of the parameters of local mesh control record.

1. AUG — Augmentation flag (0: Periodic surfaces not augmented, 1: Periodic surfaces augmented)
2. USX — Scaling factor for universe in X direction (Default — 1.0)
3. USY — Scaling factor for universe in Y direction (Default — 1.0)
4. USZ — Scaling factor for universe in Z direction (Default — 1.0)
5. ESM - Element size specification method (1: Explicit element sizes, 2: Number of elements).

If ESM is 1, the next two parameters are

a. MNS - Minimum element size in the model (specified as a linear dimension smaller than or equal to the longest dimension of the model).
b. MXS - Maximum element size in the model.

If ESM is 2, the two following parameters are

a. MXN — Maximum number of elements along the longest dimension of the model along a coordinate direction
b. MNN - Minimum number of elements along the longest dimension of the model along a coordinate direction

6. CRM - Curvature-based refinement method (0: None, 1: Method 1, 2: Method 2).

If curvature-based refinement is turned off (CRM = 0), then it must be followed by a dummy integer and a dummy real number in that order.

If method 1 of curvature-based refinement is used, the following two parameters are as follows

a. GAT - Geometric approximation type (1 - fraction of model size, 2 - explicit value)
b. GAP - Geometric approximation parameter

If method 2 of curvature-based refinement is used, the geometric approximation parameter can be specified in only one way. Therefore, the value of GAT can be any dummy integer. For this method the geometric approximation parameter, GAP, represents the geometric approximation as a fraction of the finite element edge or face size.

7. SSF - Small segment checking flag (-1: No small segment checking is performed regardless of the following records, 0: Small segment checking is performed only for those model entities specified in the following records, 1: Small segment checking is performed for all entities, except those for which this option has been specified as -1 in the following records).
8. SSR - Maximum allowable aspect ratio, i.e., maximum allowable ratio of longest edge to shortest edge for an element (> 1)
9. SFF - Elimination of small mesh entities due to small model features flag (-1: No elimination is performed regardless of the following records, 0: Only mesh entities classified on the model entities specified in the following records are considered for elimination, 1: All mesh entities not larger than SIZ (fraction of model size) are considered for elimination, except those for which this option has been specified as -1 in the following records, 2: All mesh entities not larger than SIZ (absolute) are considered for elimination, except those for which this option has been specified as -1 in the following records.

10. SIZ - Maximum size of features to be eliminated.

The parameter NLR on the third line of the meshing attributes file specifies the number of local mesh control records[5] that follow in the file.

If no local mesh controls are applied on model entities, the meshing attributes file has three lines — the first line specifies whether augmentation is on or off, the second specifies the global controls and the third contains a zero indicating that no local mesh control will be exercised (i.e. NLR should be 0).

When applying local mesh control to a model entity, each local meshing parameter must be specified on a different line for the model entity. This allows for overriding specific global meshing attributes for model entities. *Note that if a meshing attribute of a particular type is specified more than once for an entity only the last specified value will be used.* The format for a local mesh control record is shown on the fourth line of Figure 6. The meaning of the different parameters is explained below.

ENT is an integer indicating the topological entity type on which the meshing attribute is being applied and is 20 for regions, 17 for faces, 13 for edges and 11 for a vertex.

ENN is an integer representing the identifier (or number) of the model entity in the geometric modeler that Finite Octree is interfaced with. For example, for Finite Octree interfaced to Parasolid, the entity numbers will correspond to the tags of entities [2].

MAT is an integer indicates the type of mesh control that will be applied. A value of 1 for MAT indicates that element size limits will be placed on the entity, a value of 2 indicates that curvature-dependent refinement is specified for the entity and a value of 3 indicates that small segment checking will be specified for the entity.

If MAT is 1, it must be followed by the element size specification method (1 or 2). If the element size type is 1, the minimum and maximum element sizes for the entity (real numbers) must be given. If the element size type is 2, the maximum and minimum number of elements for the entity (integers) must follow. *Note — specification of limits on number of elements on individual entities is currently not supported in Finite Octree.*

If MAT is 2, then it must be followed by an integer indicating the curvature refinement method, an integer indicating the type of geometric approximation

[5] A record is assumed to be made up of a number of fields and occupies one whole line.

parameter and a real number indicating the curvature refinement parameter (with the values for the three parameters taking on the same meanings as in global mesh control.

If MAT is 3, then it must be followed by an integer indicating whether it is on (1) or off (0) for the particular entity followed by the maximum aspect ratio is for the particular entity.

If no mesh controls are specified, default global mesh parameters are used for meshing as follows

1. Augmentation of periodic surfaces — On
2. Universe shape — cubical
3. Minimum element size — $1/256^{th}$ of the model size
4. Maximum element size — $1/4^{th}$ model size
5. Curvature-based refinement — Off
6. Small segment checking — Off
7. Elimination of small mesh features due to small model features — Off

2.3 Replacing the meshing attribute retrieval operator in Finite Octree

Finite Octree gets the meshing attributes on model entities through a meshing attribute retrieval operator. If an application interfacing to Finite Octree prefers to specify attributes on model entities in a manner different from the current meshing attribute retrieval operator, it can be replaced with an equivalent operator with the same name and arguments.

The operator for retrieving meshing attributes on model entities in Finite Octree is given below

FORTRAN:

```
FT_ATBVAL(entity_type, entity_id, attribute_type,
          real_val_array, int_val_array)
```

C:

```
C_get_atbval(int entity_type, int entity,
          int attribute_type, double *real_val_array,
          int *int_val_array)
```

Input:

`entity_type` — Topological type of entity (see include file **toptype.inc** for topological type codes)
`entity_id` —Identifier of entity used by the geometric modeler
`attribute_type` — Meshing attribute types (See Section 2.2 or the include file **msattyp.inc**)

Output:

`real_val_array` — Array of real numbers depending on the meshing attribute type
`int_val_array` — Array of integers depending on the meshing attribute type

If the attribute type is T8MVAL (control of element sizes), the first integer value represents the type of element size control. If this parameter is 1 (element size control is explicit), two real values specifying the minimum and maximum element sizes should be returned in the real value array. If it is 2 (element sizes controlled by controlling the number of elements), then two more integer values indicating the minimum and maximum number of elements and no real values will be returned.

If the attribute type is T8CRVL (curvature-based refinement), the first integer in the integer values array represents the method of curvature-based refinement (0 - none, 1 - first method, 2 - second method). If method 1 is indicated, then one more integer value is returned which indicates the type of geometric approximation parameter (1 - explicit length, 2 - fraction of entity size). If method 2 is indicated, then no more integer values are returned. In both cases, the geometric approximation parameter is returned as a real number.

If the attribute type is T8SESZ (small segment checking), the first integer in the integer values array is a flag indicating whether or not small segment checking must be turned on for the entity (0 — Off, 1 — On). If the first integer is 1, then a real value representing the maximum allowable octant side length to finite element segment ratio is returned.

3 Examples

3.1 Example 1 — simple_wheel

Figure 8a shows the model of a 270° sector of a wheel of diameter 0.0167 units. The model size is equal to the diameter of the wheel. The model is very simple with planar and cylindrical surfaces only. Therefore, this model may be meshed with global mesh control on element sizes only. Although the curved surfaces of the model are candidates for curvature-based refinement, the necessary geometric approximation will be achieved simply by requesting more refinement for purposes of illustration. The meshing attributes file for the example is shown in Figure 8b and the mesh with the mesh control is shown in 8c.

Figure 9 shows an exactly equivalent mesh control for the model but with the number of elements being controlled as opposed to explicit element sizes.

3.2 Example 2 — simple_wheel meshed with local mesh control

Consider the same model as in Example 1. From the mesh in Figure 8 it can be seen that even with uniformly high refinement on the model, the mesh did not approximate the geometry of the hub very well. In this example, the element sizes on the hub face controlled below 0.003 units by overriding the global mesh control. Figure 10a shows the mesh control file. The identifier or tag of the hub face has been identified in the modeller as 162. Thus the second line indicates that local mesh control is desired for a face, the face tag is 162, element size control will be applied, the first method of element size specification (explicit sizes), the minimum element size on the face is to be 0.001 units and the maximum is to be 0.003 units.

3.3 Example 3 — simple_wheel meshed with curvature-based refinement

Consider the model of Examples 1 and 2. It was seen in those examples that the model has curved faces and therefore is a good candidate for applying curvature-based controls. In this example, curvature-based refinement will be

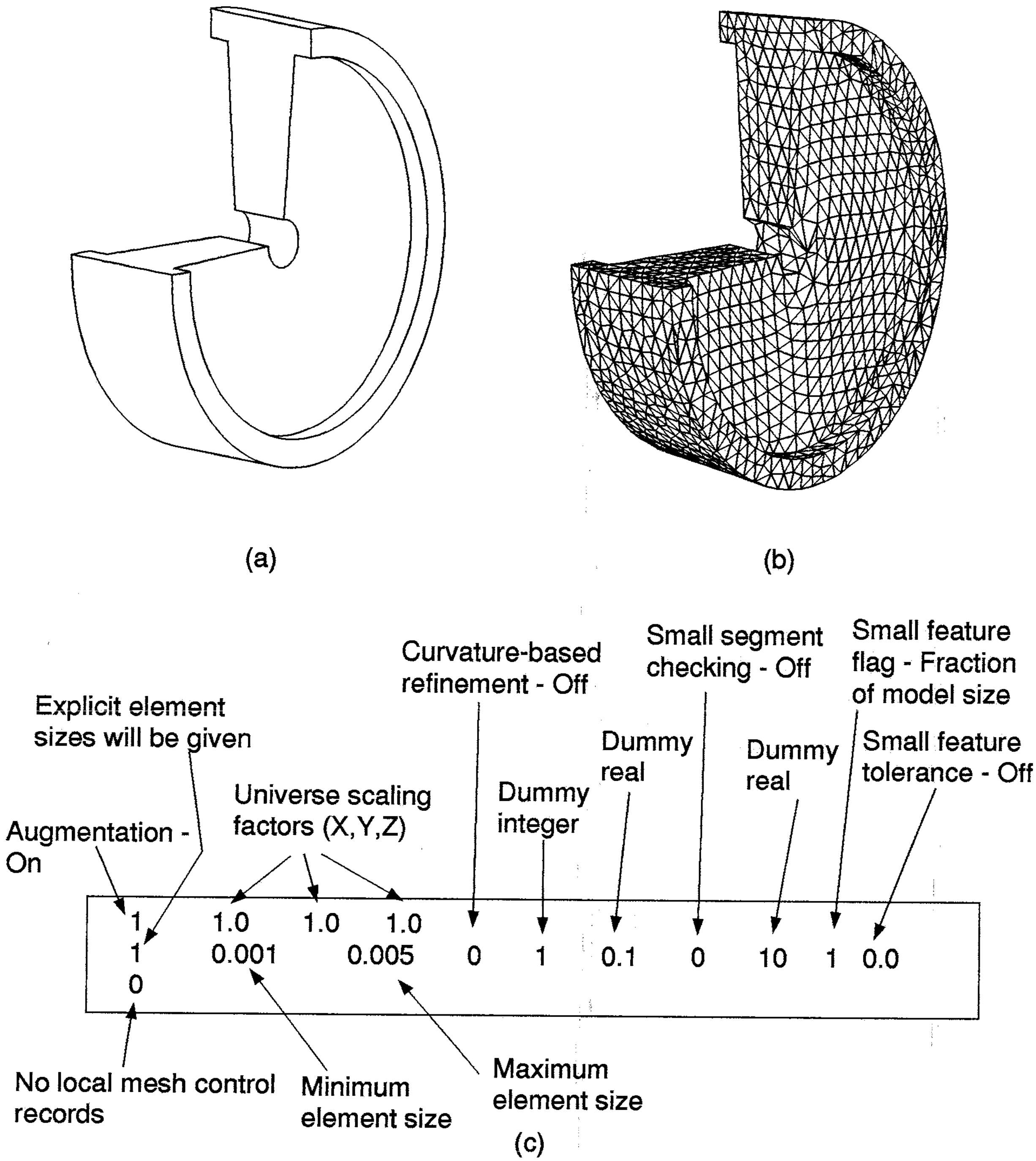


Figure 8 a) Example 1 — simple_wheel a) Model of simple_wheel b) Mesh of the model c) Meshing attributes file

1	1.0	1.0	1.0						
2	167	33	0	1	0.1	-1	10	1	0.0
0									

Figure 9 Example 1 — Element size control by specifying the maximum and minimum number of elements

applied to the model to achieve automatically achieve mesh gradations that depend on curvature.

The meshing attributes file for the model with global curvature refinement turned on is shown in Figure 11a. In the mesh control specification, the first method of curvature based refinement is used, the geometric approximation is specified as a fraction of the model size and is equal to 0.005. The mesh for the model is shown in Figure 11b

3.4 Example 4 — complex_wheel

Shown in Figure 12a is a wheel similar to the one in Examples 1–3, but with more features added such as fillets and holes. From the figure, it can be seen that the filletted corners have faces with very small width. Without any special measures taken to account for these small features, it is to be expected that elements with poor aspect ratios will be created in their vicinity. Thus, global small segment checking will be turned on for this particular model, in addition to control of element sizes and curvature-based refinement control.

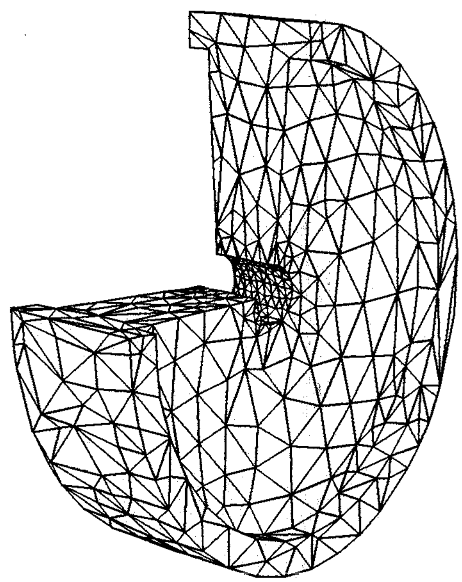
The mesh for the model is shown in Figure 12b and meshing attributes file used to generate it is shown in Figure 12c.

3.5 Example 5 — asm001

Shown in Figure 13a is a Parasolid geometric model to be meshed. The size of the model is approximately 0.24 units. The maximum element size for the mesh should be no more than 0.02 units. Since the most of the faces of the model are curved, curvature-based refinement will be turned on for the entire model. Also, it is assumed that the geometric approximation error on curved faces A and B must be less than 1/1000th the model size and that on face C less than

0.00005 with a minimum element size limit of 0.005 units. Lastly, small segment checking is turned on for the faces D and E.

The meshing attributes file is shown in Figure 13b. The first line of the file indicates the following controls



(a)

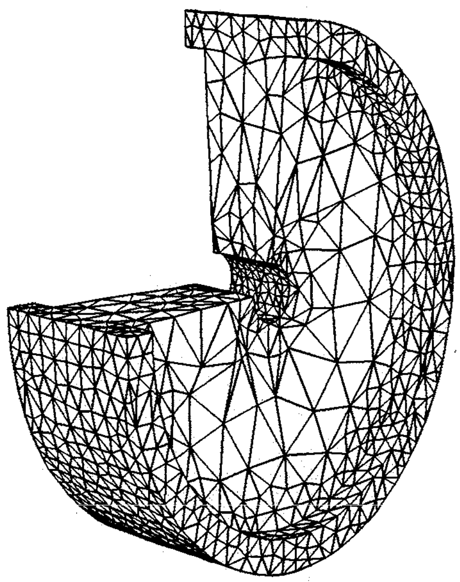
```
1    1.0     1.0     1.0
1    0.001   0.01    0    1    0.2    -1    10   1   0.0
1
17   162   1   1   0.001   0.003
```

(b)

Figure 10 Example 2 — a) simple_wheel meshed with local mesh control b) meshing attributes file

```
1  1.0  1.0  1.0
1  0.0005  0.01   1  1  0.0005   -1  10 1 0.0
0
```

(a)



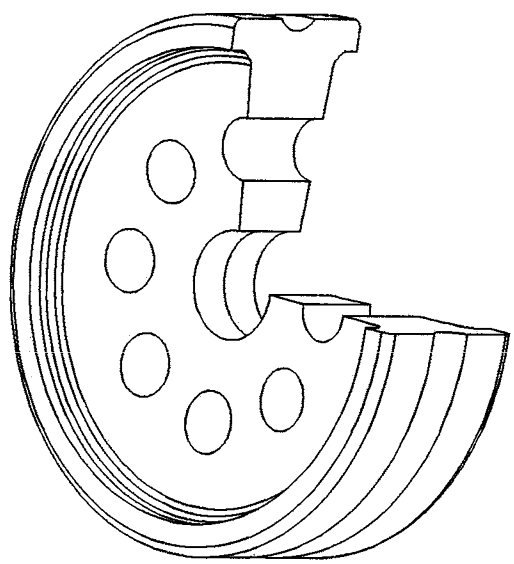
(b)

Figure 11 Example 3 — Global curvature refinement on simple_wheel
a) Mesh with curvature-based refinement b) Meshing attributes file

1. The first method of element size specification will be used, the minimum element size 0.005 units, the maximum element size is 0.02 units
2. The second method of curvature-based refinement will be applied to all model entities with the geometric approximation error to finite element edge length ratio maintained below 0.1, minimum element size permitting.
3. Small segment checking will not be applied to all entities of the model

The second line of the file indicates that local mesh control will be applied on 5 model entities.

The next three lines apply curvature-based refinement in different ways to the faces A, B and C. For example, line 3 indicates that the mesh control will be applied on a face (ENT = 17), the entity identifier (obtained from Parasolid, v5.3) is 79 (face A), curvature-based refinement is requested for this face, the first method of curvature-based refinement is to be used, the geometric approximation error will be specified as a fraction of the model size (GAT = 1) as 0.001. Line 4 applies exactly the same control to face B in a different manner. Curvature-based refinement is also applied to face C on line 5 with a lower geometric



(a)

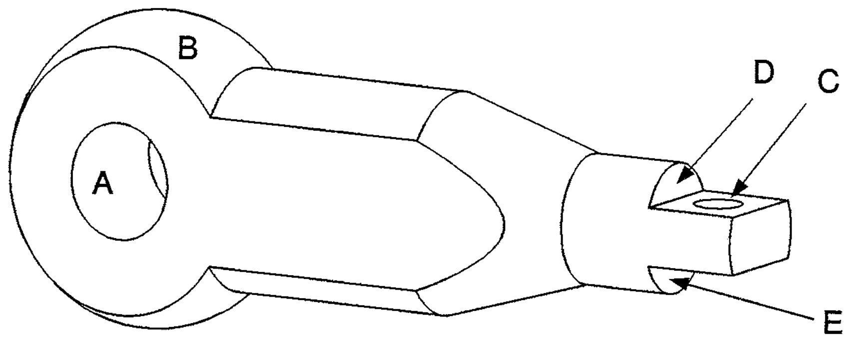
(b)

```
1  1.0   1.0   1.0
1  0.0005  0.01   1   1   0.0005   1   10  1  0.0
0
```

(c)

Figure 12 Example 4 — complex_wheel a) geometric model b) mesh of model c) meshing attributes file approximation error. Lines 6 and 7 apply small segment checking to faces D and E and specify that elements with edges classified on these faces should have aspect ratios below 10.

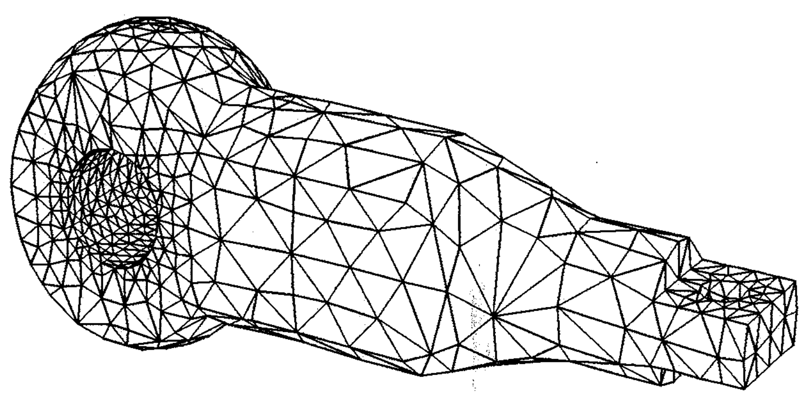
The mesh for this model with the given controls is shown in Figure 13c



(a)

```
1   1.0    1.0    1.0
1  0.005  0.02  2  1  0.1  0  10  1  0.0
5
17   79  2  1  1  0.001
17  128  2  1  2  0.00024
17   22  2  1  2  0.00005
17   83  3  1  10
17   87  3  1  10
```

(b)



(c)

Figure 13 Example 5 a) Geometric model
b) Meshing attributes file c) Mesh of model

4 Bibliography

[1] Saikat Dey. Curvature sensitive refinement in 3d automatic mesh generation. Master's thesis, Scientific Computation Research Center, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, August 1993.

[2] Shape Data Limited, Parker's House, 46 Regent Street, Cambridge CB2 1DB England. *PARASOLID v4.0 Programming Reference Manual*, August 1991.

[3] M. S. Shephard and M. K. Georges. Automatic three-dimensional mesh generation by the Finite Octree technique. *Int. J. Numer. Meth. Engng.*, 32(4):709–749, 1991.

[4] Mark S. Shephard, Marcel K. Georges, and Hugues L. de Cougny. Geometric modeling interactions required to support a fully automatic mesh generator. Technical Report SCOREC # 25-1993,, Scientific Computation Research Center, Rensselaer Polytechnic Institute, Troy, NY 12180-3590, 1993.