Automatic Hexahedral Mesh Generation with Feature Line Extraction

Masayuki Hariya[1], Ichiro Nishigaki[2], Ichiro Kataoka[3], Yoshimitsu Hiro[4]

1, 2, 3 Mechanical Engineering Research Laboratory, Hitachi, Ltd.
832-2, Horiguchi, Hitachinaka, Ibaraki, Japan
[1]masayuki.hariya.sa@hitachi.com
[2]ichiro.nishigaki.mp@hitachi.com
[3]ichiro.kataoka.pq@hitachi.com
[4] Industrial Information System Division, Hitachi, Ltd.
Omori Bellport B Bldg., Minami Oi 6-chome, Shinagawa-ku, Tokyo, Japan
yoshimitsu.hiro.yc@hitachi.com

Abstarct. An improved method using feature line extraction is described for automatically generating hexahedral meshes for complex geometric models that automate the normally interactive operations (such as model editing). Testing showed that the time taken for these interactive operations was significantly reduced, making it possible to quickly generate hexahedral meshes with sufficient quality for complex models. Application of this method to mechanical part models showed that it shortened the time to generate a mesh in about 10% the time required with the previous method.

1. Introduction

The use of digital engineering in manufacturing to shorten product-development times is becoming more widespread. Both three-dimensional computer-aided-design (3D-CAD) modeling and computer-aided engineering (CAE) are widely used by product designers. A promising approach to reducing the CAE preprocessing time is automatic mesh generation [1].

Hexahedral meshes are widely used in numerical analyses because they generally provide more accurate results than tetrahedral meshes. They are usually generated by hand by dividing a complex geometric model into simple geometrical blocks. As this requires many interactive procedures, various methods for automatically generating them have been proposed [2–6].

One is based on the Octree method [2]. It generates small cubes inside the geometric model and generates a mesh by mapping the surfaces of the cubes on the boundary onto the surfaces of the geometric model. Another method automates the division of the blocks [3–6].

Our previous method for automating the process is based on shape recognition and boundary fitting [7–8]. A unique shape-recognition technique is used to change a geometric model into an approximate one consisting of straight lines. Boundary fitting maps small cubes that are generated by dividing the approximate model, onto the geometric and generate hexahedral meshes. This reduces the number of interactive procedures, and thus reduces the time to generate a hexahedral mesh by about 60%. However, this sometimes can not generate meshes automatically in case of some complicated models. We have now improved our method by also automating the model-editing task by using feature line extraction.

2. Previous Mesh Generation Method

2.1 Shape Recognition and Boundary Fitting

The procedure for generating hexahedral meshes using shape recognition and boundary fitting is illustrated in Figure 1.

Step 1-1: An approximate model (Figure 1(b)), called a recognition model, is constructed from a geometric model (Figure 1(a)) by using only the lines parallel to the Cartesian axes. The recognition model must be topologically identical and geometrically similar to the geometric model.

Step 1-2: The edge lengths of the recognition model are adjusted to the nearest integral multiple of the standard element sizes, and the model is divided into cubes, producing what is called a mapping model (Figure 1(c)).

Step 1-3: The mapping model is mapped onto the original geometric model by boundary fitting, generating the final hexahedral mesh (Figure 1(d)). The input data is the standard element size.

The shape recognition is based on fuzzy-logic theory. All edges of the geometric model are arranged parallel to the Cartesian coordinates (ξ,η,ζ) of the mapping space. The axis where an edge is arranged is determined based on information obtained from the geometric model, such as the

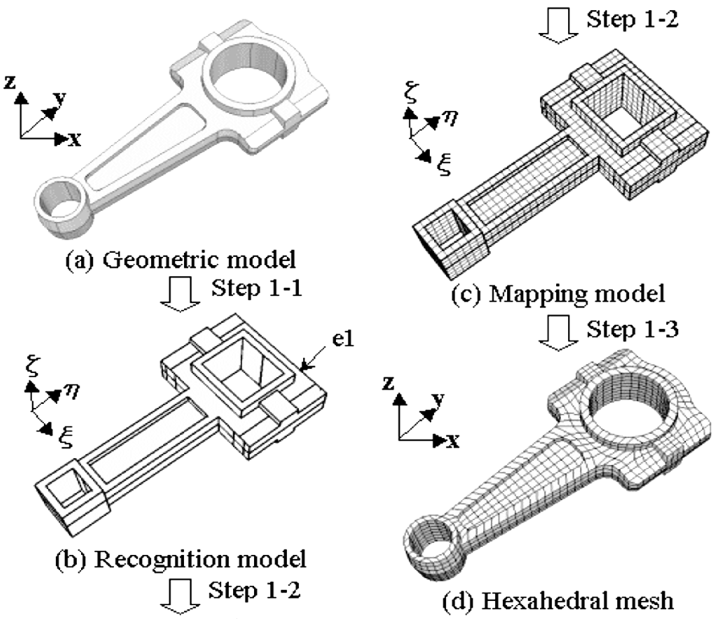


Figure 1 Procedure for generating hexahedral meshes using shape recognition and boundary fitting

direction vectors of the edges and the vertex angle of two edges. The directions of the coordinate axes are called the assigned directions. The shape of the recognition model is determined from the assigned directions of the edges. For example, in Figure 1(b), the assigned direction of edge e1 is the ξ axis.

2.2 Problems

With this mesh generation procedure, two typical errors may result in a failure to generate a recognition model.

(1) If the geometric model contains surfaces where the boundary has three or fewer edges, a recognition model that is topologically equal to the geometric model cannot be generated.

(2) If the assigned edge directions are not correct, a recognition model cannot be generated even if the geometric model is topologically correct.

Interactive operations (adding or deleting a line to or from the geometric model, correcting an assigned direction, etc.) are necessary in such cases before the recognition model can be generated.

3. Improved Mesh Generation Method

Our improved method avoids the two problems described above by using feature line extraction to generate a recognition model. The procedure for generating a recognition model is illustrated in Figure 2.

Step 2-1: Triangular meshes (Figure 2(b)) are generated on the surface of the geometric model [Figure 2(a)].

Step 2-2: A feature-shape model [Figure 2(c)] is generated by extracting the feature lines from the boundaries of the triangular meshes [Figure 2(b)].

Step 2-3: The recognition model [Figure 2(d)] is generated from the feature-shape model.

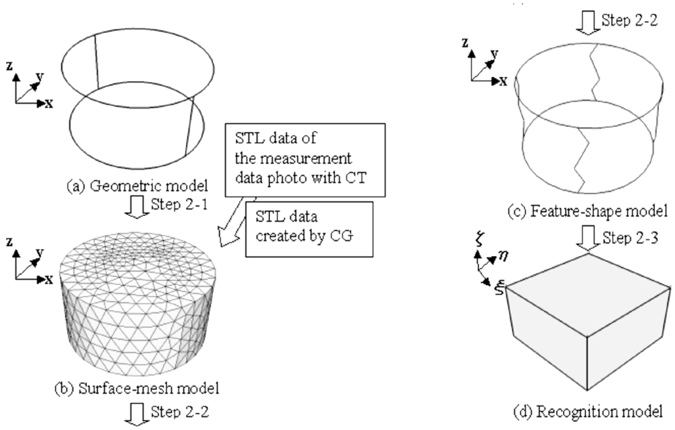


Figure 2 Procedure for generating recognition model using feature line extraction

This improved method has two key features. First, the feature lines used for generating the recognition model can be automatically extracted from the triangular meshes on the geometric model. This means that the generation of the recognition model does not depend on the topology of the geometric model. Therefore, a hexahedral mesh can be generated without having to manually divide a complex geometric model into simple geometrical blocks. Second, the feature lines are selected by taking the assigned directions into consideration using a method that will be described below. This means that a hexahedral mesh can be generated without correcting the assigned directions.

Furthermore, STL data of the measurement data photo with computed tomography (CT) equipment and STL data created by a computer graphics (CG) program can be used for the surface-mesh model (Figure 2(b)). This enables the use of finite-element analysis, which uses a high-quality hexahedral mesh about a wide range of steps of the engineering.

As shown in Table 1, the current method (based on feature line extraction) is better than the previous one (based on fuzzy logic) in terms of the automation of mesh generation and the application range, while the previous method is better in terms of the generation of meshes that represent the geometric model. This is because, in the current method, the shape of the generate meshes is affected by the surface meshes used.

Table 1. Comparison between mesh generation methods

	Previous	Current
Automation of mesh generation	average	good
Representation of geometric model	good	average
Application range	wide	very wide

3.1 Generating Recognition Model

The recognition model is generated as shown in Fig. 3.

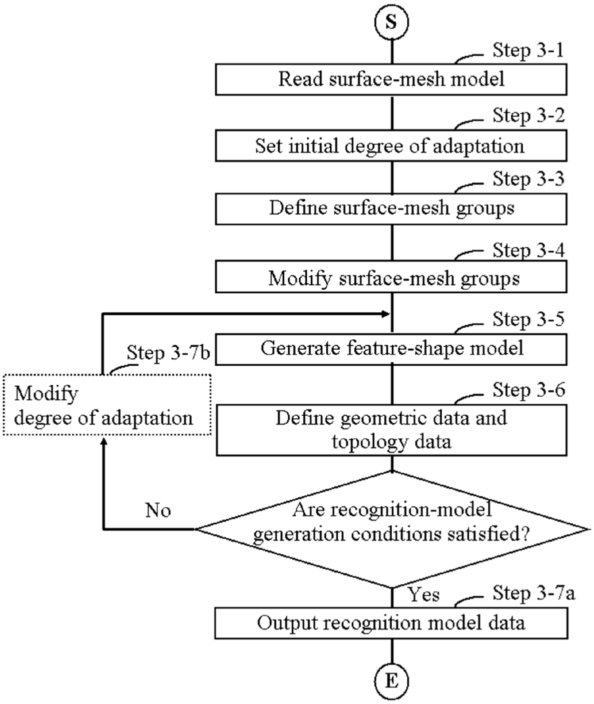


Figure 3 Steps in generating recognition model

3.1.1 Read Surface-Mesh Model: Step 3-1

The recognition-model generation program reads the surface-mesh model and registers it in a data table. The mesh corresponding to the surface-mesh model is called the mesh face, the boundaries of the mesh face are called the mesh lines, and the nodal points of the mesh face are called mesh points.

3.1.2 Set Initial Degree of Adaptation: Step 3-2

Unit normal vector v = (vcx, vcy, vcz) is set as the initial degree of adaptation $P = (P_\xi, P_\eta, P_\zeta)$ for each mesh face. This degree is what a mesh face is assigned in the direction of a particular coordinate axis (ξ,η,ζ) in the mapping space. The normal vector can be calculated from the coordinates of the mesh points composing the mesh face.

3.1.3 Define of Surface-Mesh Groups: Step 3-3

Surface mesh groups are defined by grouping the mesh faces based on their degrees of adaptation.

First, the assigned directions of the mesh faces are determined: P_ξ, P_η, and P_ζ are compared, and the assigned directions are taken as the maximum absolute values of the axial directions. The assigned directions are modified based on the signs of the degrees of adaptation. For example, if $P_\xi = 1.0$, $P_\eta = 0.0$, and $P_\zeta = 0.0$, the initial assigned direction is the ξ axis. The actual direction is $-\xi$ because the sign of P_ξ is minus.

Next, the mesh faces with the same assigned direction and sharing a mesh line are defined as a surface- mesh group. The assigned direction of the group is the assigned direction of the mesh faces composing it.

Example surface-mesh groups are shown in Figure 4. Mesh faces with the same shading represent one surface-mesh group. The assigned direction of surface-mesh group 1 (Gr1) is $+\xi$, based on the assigned direction of mesh face MF1.

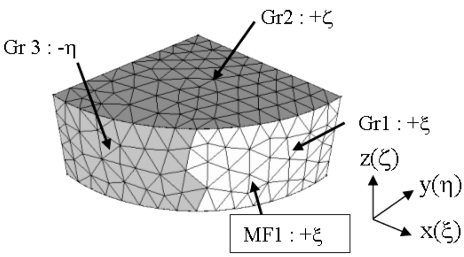


Figure 4 Surface-mesh groups

3.1.4 Modify Surface-Mesh Groups: Step 3-4

The defined surface-mesh groups are modified based on group correction rules, which are necessary conditions for relating two or more surface-mesh groups. Here we consider two example conditions.

Condition 1: A surface-mesh group must have four or more neighboring surface-mesh groups.

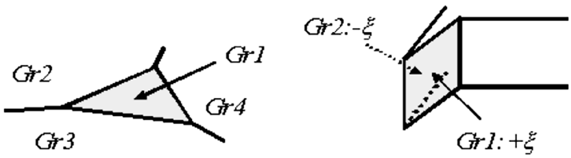
Condition 2: The assigned direction of a surface-mesh group should not be in the same axial direction as the assigned direction of a neighboring surface-mesh group.

Example surface-mesh groups not satisfying these conditions are shown in Figure 5. In Figure 5(a), Gr1 touches only three surface-mesh groups. In Figure 5(b), Gr1 and Gr2 are both assigned the ξ-axis direction.

An example of modifying a surface-mesh group to satisfy Condition 1 is shown in Figure 6. First, the length of the line shared by each pair of surface-mesh groups is calculated. The shared line length for Gr1 and Gr2 is L2, the shared line length for GR1 and Gr3 is L3, and the shared line length for Gr1 and Gr4 is L4. A surface-mesh group that has three or less neighboring surface-mesh groups is combined with the one with which is has the longest shared line, and a new surface mesh group is generated. Because L3 is the longest shared line for Gr1, Gr1 is combined with Gr3, creating Gr5.

An example of modifying a surface-mesh group to satisfy Condition 2 is shown in Figure 7. One of the two groups with the same assigned direction is divided in two and setting a new assigned direction. In the figure, Gr5 is created from Gr1 and is assigned the direction of the ζ axis because the assigned directions of the neighboring surface-mesh groups are ξ and η. The actual assigned direction is thus $-\zeta$ based on the z component of the average normal vector of the mesh faces composing Gr5.

Many other group correction rules are used besides the two described above.



(a) Condition 1: Not satisfied (b) Condition 2 : Not satisfied

Figure 5 Irregular-mesh groups

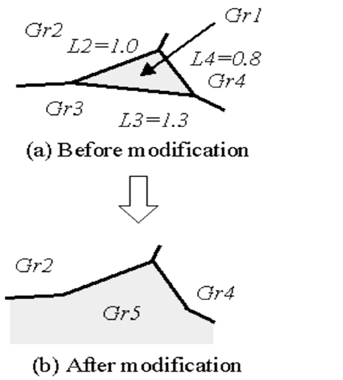


Figure 6 Group correction rule to satisfy Condition 1

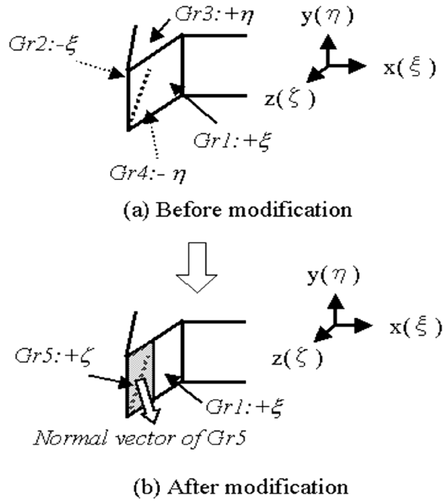


Figure 7 Group correction rule to satisfy Condition 2

3.1.5 Generate Feature-Shape Model: Step 3-5

Feature surfaces are defined as the defined and then modified surface-mesh groups, and their assigned directions are the assigned directions of the groups. A feature line is defined as a set of mesh -lines shared by two different surface-mesh groups. The set of mesh lines, ML1–4, for the surface-mesh model in Figure 8 comprising the boundary between GrA and GrB is defined as one feature line.

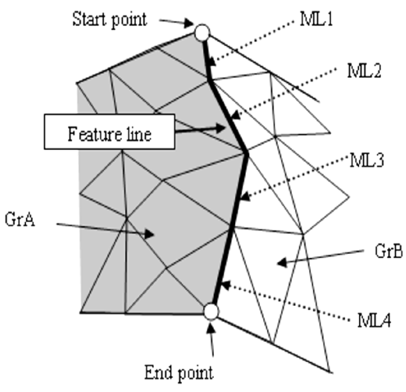


Figure 8 Feature line

The assigned direction of a feature line is determined from the combination of the assigned directions of the two surface-mesh groups to which the feature line belongs. The relation between the assigned directions of the surface-mesh groups and the assigned direction of the feature line is shown in Table 2.

Table 2 Assigned direction of feature line

Assigned direction of surface-mesh group A	Assigned direction of surface-mesh group B	Assigned direction of feature line
ξ	η	ζ
η	ζ	ξ
ζ	ξ	η

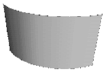
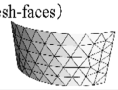
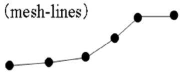
The sign of the assigned direction is the sign of the direction vector of the feature line. For example, if the assigned direction is ξ and the sign of the x value of the direction vector is minus, the assigned direction is modified to $-\xi$.

3.1.6 Define Geometric Data and Topology Data: Step 3-6

The data for the feature-shape model are defined using boundary representation, which is expressed using topology data showing the connection between geometric elements and using geometric data showing the shape of each geometric element.

The correspondence between the geometric model and the feature-shape model is shown in Table 3. The geometric data for the feature points are the 3-D coordinates of the points. Moreover, the feature surface is composed of mesh -faces, and the feature line is composed of mesh -lines. The topology data has a hierarchical structure like the geometric model.

Table 3 Correspondence between geometric model and feature-shape model

	Geometric model	Feature-shape model
2nd-order geometric element	Surface 	Feature face (mesh-faces) 
1st-order geometric element	Line 	Feature line (mesh-lines) 
0th-order geometric element	Point	Feature point (mesh-point)

3.1.7 Check Feature-Shape Model and Dssigned Directions

To generate the recognition model, the feature-shape model and assigned directions must satisfy the following recognition-model generation conditions.

<Condition 1> At least one feature surface is assigned the plus or minus direction of one of the (ξ,η,ζ) axes.

<Condition 2> Neighboring feature surfaces are not assigned to the opposite direction of same axial direction.

<Condition 3> There must be at least one feature line comprising the feature surface boundary, which is assigned the plus or minus direction of two coordinates axes.

<Condition 4> Neighboring feature lines are not assigned to the opposite direction of same axial direction.

Example satisfied and unsatisfied conditions are shown in Figure 9.

If the conditions are satisfied, Step 3-7a is carried out. If the conditions are not satisfied, Step 3-7b is carried out.

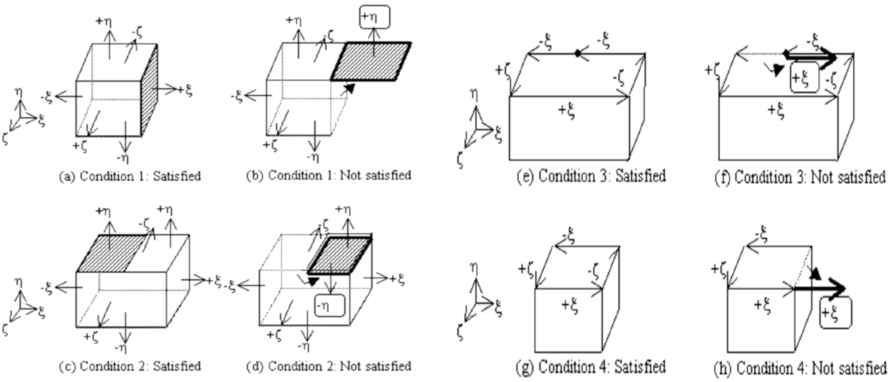


Figure 9 Satisfied and unsatisfied recognition-model generation conditions

3.1.8 Output Recognition Model Data: Step 3-7a

The feature-shape model and assigned directions are registered in the database.

3.1.9 Modify Degree of Adaptation: Step 3-7b

The degree of adaptation of the mesh face (MF) is modified based on the degrees of adaptation of the neighboring mesh faces. The number of neighboring mesh faces is denoted by n. The degree of adaptation of a MF before modification is denoted by $P_b = (P_{\xi b}, P_{\eta b}, P_{\zeta b})$, and the degree of adaptation after modification is denoted by P = (P_ξ, P_η, P_ζ). The degree of adaptation before correction of neighboring mesh -faces MF(i): (I = 1,2,…,n) is denoted by P_{bi} = ($P_{\xi bi}$, $P_{\eta bi}$, $P_{\zeta bi}$), and the degree of adaptation of a MF after modification is calculated as follows.

$$\begin{cases} P_\xi = P_{\xi b} + \dfrac{\beta}{n}\sum_{i=1}^{n} P_{\xi bi} \\ P_\eta = P_{\eta b} + \dfrac{\beta}{n}\sum_{i=1}^{n} P_{\eta bi} \\ P_\zeta = P_{\zeta b} + \dfrac{\beta}{n}\sum_{i=1}^{n} P_{\zeta bi} \end{cases} \tag{1}$$

$$\begin{cases} P'_\xi = P_\xi \\ P'_\eta = P_\eta \\ P'_\zeta = P_\zeta \\ val = \sqrt{P_\xi \times P_\xi + P_\eta \times P_\eta + P_\zeta \times P_\zeta} \\ P_\xi = P'_\xi / val \\ P_\eta = P'_\eta / val \\ P_\zeta = P'_\zeta / val \end{cases} \tag{2}$$

In the equations, β is a parameter reflecting the speed of modification. Equation (1) is used to smooth the degree of adaptation, and equation (2) is used to normalize the degree of adaptation calculated using equation (1). Small surface-mesh groups are removed at the time of a surface mesh-group definition by modification, as described above. As a result, the generation of the recognition model is simplified.

4. Examples of Generated Meshes

We extracted feature lines to generate a hexahedral mesh of a mechanical-part model (Figure 10(a)) for use in structural analysis. Figure 10(b) shows the surface-mesh model used to generate the mesh. The feature-shape model (Figure 10(c)) was generated by extracting the feature lines. Not all

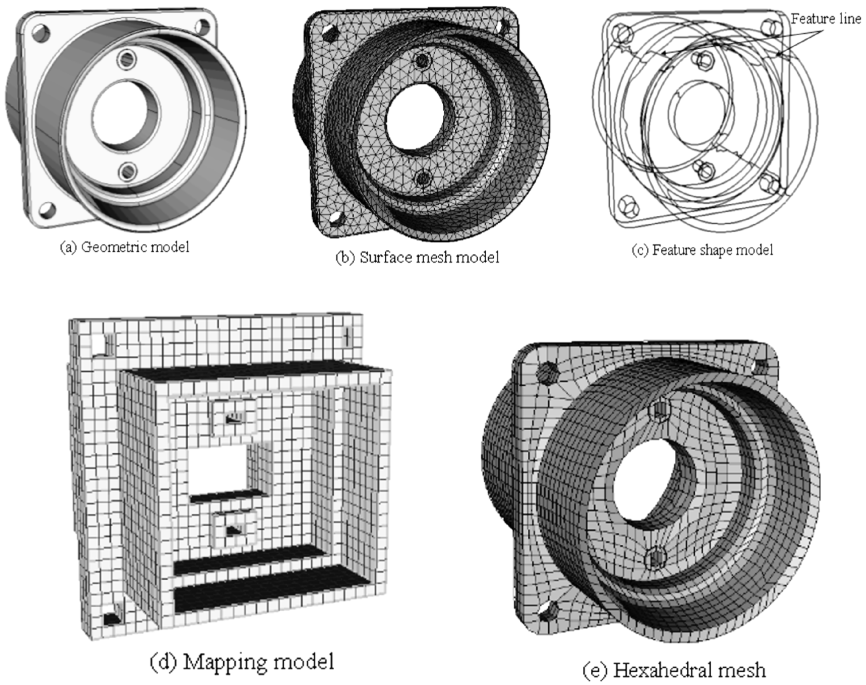


Figure 10 Procedure for generating hexahedral mesh for mechanical part

the feature lines corresponded to the ones in the geometric model. Figure 10(d) shows the mapping model, and Figure 10(e) shows the generated hexahedral mesh.

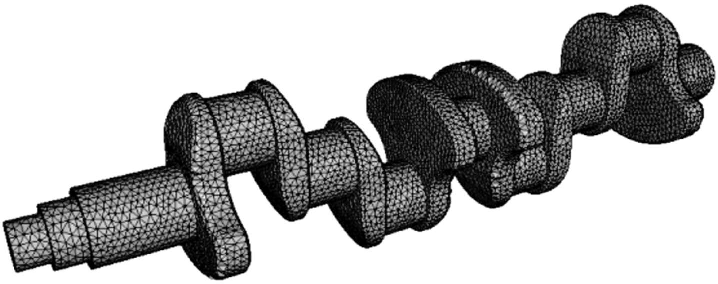
Table 4 summarizes the mesh-generation times for the previous and current methods. The current method requires no time for interactive operations, and its calculation time is about half that of the previous one because there is less trial and error. As a result, its mesh generation time is about 10% that of the previous one.

Two other examples are shown in Figures 11 and 12. The crank-shaft (Figure 11(b)) has 8200 elements, and the connecting rod (Figure 12(b)) has 1788. The calculation time for the former was 30 sec, and that for the latter was 5 sec. In both cases, the hexahedral mesh was automatically generated without interactive operations, such as line creation necessary for recognition model.

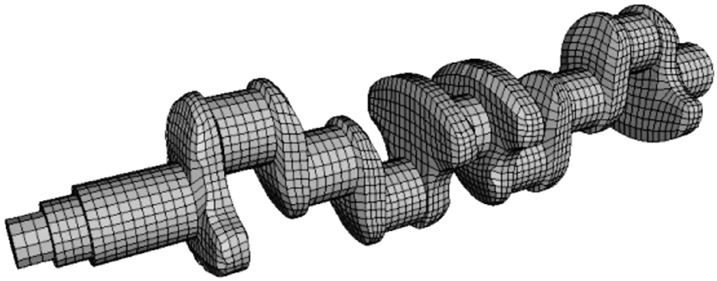
Our improved method does not require the operator to perform any skillful interactive operations such as block division. That is, someone without any specialized knowledge can generate hexahedral meshes.

Table 4 Comparison of mesh generation times (min)

	Previous method	Current method
Calculation	5	2
Interactive operation	15	0
Total	20	2

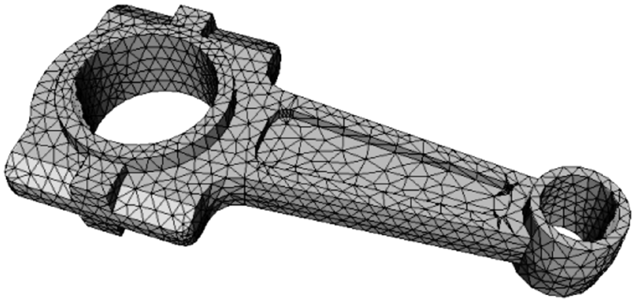


(a) Surface-mesh model

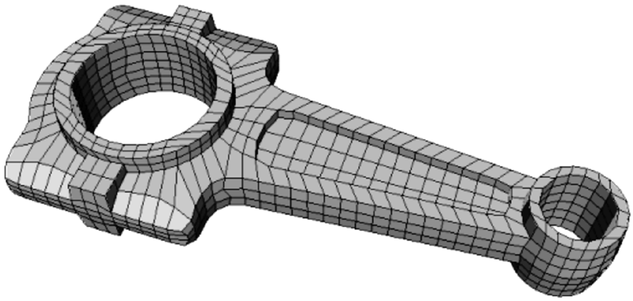


(b) Hexahedral mesh

Figure 11 Surface-mesh model and hexahedral mesh for crank shaft



(a) Surface-mesh model



(b) Hexahedral mesh

Figure 12. Surface-mesh model and hexahedral mesh for connecting rod

5. Conclusion

Our improved method for automatically generating hexahedral meshes can be used to reduce the number of interactive operations needed to generate meshes. Application to complex geometric models showed that it can

(1) automatically extract feature lines from the boundary between surface meshes, enabling a recognition model to automatically be generated and a hexahedral mesh to be quickly generated, and

(2) generate a hexahedral mesh of a mechanical-part model in about 10% the time required with the previous method.

References

1. Takamura, A. Kitani and T. Sasaki, “Automatic mesh generation for vehicle aerodynamics analysis”, VDI-Berichte, (1998) 979–988
2. R. Schneiders, “Octree-based generation of hexahedral element meshes”, Proc. 5th Int. Meshing Roundtable '96, SAND 96-2301 (1996) 205–215
3. T. Blacker, “The Cooper tool”, Proc. 5th Int. Meshing Roundtable '96, SAND 96-2301 (1996) 13–30
4. B.Y. Shin and H. Sakurai, “Automated hexahedral mesh generation by swept volume decomposition and recomposition”, Proc. 5th Int. Meshing Roundtable '96, SAND 96-2301 (1996) 273–280
5. R. Taghavi, “Automatic block decomposition using fuzzy logic analysis”, Proc. 9th Int. Meshing Roundtable '00, (2000)
6. N. A. Calvo and S. R. Idelsohm, “All-hexahedral element meshing: Generation of the dual mesh by recurrent subdivision”, Comput. Methods Appl. Mech. Engrg. 182 (2000) 371–378
7. H. Takahashi and H. Shimizu, “A general purpose automatic mesh generation using shape recognition technique”, Compt. Engrg. ASME 1 (1991) 519–526
8. N. Chiba, I. Nishigaki, Y. Yamashita, C. Takizawa, and K. Fujishiro, “A flexible automatic hexahedral mesh generation by boundary-fit method”, Comput. Methods Appl. Mech. Engrg. 161 (1998) 145–154