42

Application of Cartesian Grid to Flow Computation around Supersonic Transport

Paulus R. LAHUR[*] and Yoshiaki NAKAMURA[†]
Dept. of Aerospace Eng., Nagoya University

Key Words: Grid Generation, Cartesian Grid, Thin Body, Sharp Edge, Cell Splitting

Abstract

A Cartesian grid generation method has been developed for fast and automated analysis of inviscid flows around 3D bodies. It can take into account non-simple cases of intersection between a solid surface and a Cartesian grid, so that a cell containing a thin body, a sharp edge or multiple solid regions is properly handled. These cases appear frequently in the computation of a supersonic transport (SST) model. To increase the efficiency of computation, a pseudo-planar approximation is made to represent a body surface inside a cell, resulting in a reduction of total body surface elements by a factor of about three in this application. Higher efficiency is achieved using a new algorithm for local grid refinement around a curved surface.

1. Introduction

Cartesian grids have become popular recently, due to their ability to treat complicated geometries with less effort. The time of grid generation is also very short. However, being non-body-fitted, Cartesian grids may have problems concerning intersection between grid and solid surface. One of such problems is found in handling a very thin part of a body, such as the outboard wing of SST, where a cell might contain two flow regions separated by the thin body, each of which belongs to a different wing surface: upper and lower surfaces. If no distinctions between those two flow regions are made, a conventional flow solver will mistakenly consider them as one flow region, leading to an erroneous solution.

There are several possibilities to overcome this problem. (1) Multiple overlapping grids can be locally fitted to some parts of the body.[1] However, this grid generation process can no longer be made automatic nor simple. (2) The problem cell can be refined until each cell contains a single flow region.[2-4] This approach is impractical as a large number of cells are produced and the cell size becomes prohibitively small. (3) It is a hybrid of Cartesian and prismatic grids. The latter grid is employed to discretize a boundary layer, which intersects with a Cartesian grid outside it.[5-6] However, this grid generation process would be even more complicated. (4) The problem cell can be split into two sub-cells in such a way that each sub-cell contains a unique flow region,[7,9-11] as introduced by Melton, Berger, Aftosmis and Wong.[7]

An alternative cell-splitting algorithm is proposed here in this study, where intersection between a body surface and a cell is simplified using pseudo-plane approximation to increase computational efficiency.[8] Validation has been carried out on a very thin double wedge in a supersonic flow, where significant improvement in accuracy was achieved. The method can be further extended to consider a cell with a sharp edge as well as a cell with two solid regions, the cases of which are found when treating a SST model. A new local grid refinement around a curved surface is also proposed for a better efficiency.

2. Grid Generation Method

The procedure of the grid generation method proposed here is briefly outlined in the following. First a coarse, uniform grid is generated around a body inside a computational domain. Then grid refinement is made around the body surface in such a way that the grid resolution increases toward the surface. Further refinement is performed around curved parts of the body.

Then we compute the geometrical properties of cells with irregular shape produced by intersection with the body surface. For this, a special algorithm has been developed in this study, so that even complicated case of intersection can be treated.

2.1 Treatment of Irregular Cells at Body Surface

Generally, intersection between a grid cell and a body surface has many levels of complexity, which can be classified by the number of fluid or solid regions existing inside a cell (see Fig. 1). Class 1 is the case where there is no intersection with body surface, which is the simplest case. Class 2 is a simple intersection, where there are one flow region and one solid region within the cell. When the intersection becomes more complicated, there might be two flow regions and one solid region, or two solid regions and one flow region within the cell, which belongs to class 3. The present method can treat the cases ranging from classes 1 to 3. Beyond these three cases, intersection becomes too complicated to deal with, and furthermore, such cases

[*] Graduate Student, Graduate School of Engineering
[†] Professor, Department of Aerospace Engineering
Nagoya University, Furo-cho, Chikusa-ku, Nagoya 464-8603

are so rare, that the cell is further refined.

The intersection cases mentioned above determine calculation methods of the cell geometrical properties, which consist of (1) volume, (2) face area, and (3) face normal vector. The outline of the algorithms used here is shown in Fig. 2.

Important features of the algorithms, such as pseudo-plane approximation, and the treatment for cells intersecting body surface, are briefly described in the following sub-sections.

2.1.1 Pseudo-Plane Approximation

The following two assumptions are a base of the method proposed in this study:

1. A body surface is approximated to be pseudo-planar within a cell (see Fig. 3).
2. One cell at body surface contains one solid region and one fluid region.

The first assumption simplifies the representation of body surface within a cell by a single panel, and thus minimizes the requirement for memory space. The term 'pseudo' is used since the actual body surface within a cell is not planar in general. Its normal vector and area are thus calculated from the constituent triangular panels (see Figs. 3b and 3c). In other researches dealing with the cell-splitting method, such simplification is not made, so that all body elements inside a cell are stored, as shown in Fig. 3a.[7,9-11] Therefore, those methods generally need a larger memory space than the present one.

The second assumption provides another simplification since the relation between cell and body surface is made unique, leading to efficient computation.

2.1.2 A Cell Containing A Normal Geometry or A Sharp Edge (Class 2)

To classify an intersection, the number of body surface within a cell is first identified (see Figs. 2 and 4). When there is only one body surface, the body geometry is classified as either normal geometry or sharp edge. Here a sharp edge is defined as a pair of neighboring body panels with $\theta < 30°$, where θ is the angle made between these panels.

For the case of normal geometry, computing cell geometrical properties is simple, since a polyhedron representing the flow region inside the cell can readily be constructed. For the sharp edge case, however, according to the first assumption mentioned above, the body inside the cell will be truncated, or blunted (see Fig. 5a). This is undesirable since it deteriorates solution accuracy. To preserve the original shape as much as possible, the cell should be split along the sharp edge by placing a splitting panel (see Fig. 5b). By considering each flow region separately, the geometrical properties can be computed in the same manner as the normal case.

2.1.3 A Cell Containing Two Flow Regions or Two Solid Regions (Class 3)

A cell contains two body surfaces in the following two cases: (1) one solid region and two flow regions, and (2) two solid regions and one flow region (see Fig. 1). Two solid regions exist in a cell if the surfaces are facing each other, that is, Eqn. (1) is satisfied (see Fig. 6).

$$\hat{n}_1 \bullet \left(\vec{X}_{c_2} - \vec{X}_{c_1}\right) > 0, \quad \hat{n}_2 \bullet \left(\vec{X}_{c_1} - \vec{X}_{c_2}\right) > 0 \qquad (1)$$

where $\vec{X}_C$ and $\hat{n}$ are the position vector of centroid and the normal vector of the pseudo planar body surface, respectively. Subscripts 1 and 2 denote each body surface.

In both cases, the cell is split into two sub-cells, where only one interface between fluid and solid regions is considered in the cell, following the second assumption that there can only be one flow region and one solid region in a cell. The geometrical properties of polyhedron of each flow region are then computed as in the case of normal cell geometry.

Although in case (2) the cell should not be split since it contains only one flow region, it is actually split to be consistent with the second assumption, where the geometry properties in each sub-cell are modified so as to become the same as those of the original cell. They are re-merged in the stage of solving to give an identical solution equivalent to the original one (see Fig. 2).

2.2 Local Grid Refinement around Curved Body Surface

Here in this study grid refinement is carried out isotropically, which means that a three-dimensional cell is refined into eight sub-cells of equal shape and size in all Cartesian directions.

Since the pseudo-planar approximation is less accurate at curved body surface, more refinement is needed as the curvature of body surface increases. However, as the curvature becomes infinity at a sharp edge, excessive refinement will result, and consequently a large number of very small cells will be produced. In this study, such cell is split instead. Thus, grid refinement criteria used by other methods cannot be applied, because more refinement takes place as the body surface becomes more curved. Those criteria are listed in the following; (1) a cell larger than the characteristic size of body panels it intersects,[5,6] (2) small angle between neighboring body pânels within a cell,[7-10] and (3) a cell with more than one vertex of body panel.[11]

A new criterion is proposed here, where local grid refinement is carried out between two extreme cases: sharp edge and flat surface. To implement this, the absolute value of the cosine of angle θ between two body panels is employed, since it takes a maximum value of 1 at $\theta = 0°$ (sharp edge) and 180° (flat surface), and a minimum value of 0 at $\theta = \pm 90°$, as in Eqn. (2). The minimum value of all combinations of body panels within the cell is considered to capture a curved body surface even if its mesh is so fine that the angle between neighboring panels tends to that of a flat surface.

An edge with $\theta < \theta_{lim} = 30°$ is considered a sharp edge in this study, which is treated by the method described in section 2.1.2. On the other hand, a cell containing body panels with 30° ≤ |θ| ≤ 150° is refined.

$$\min\left|\hat{n}_i \bullet \hat{n}_j\right| < \cos\theta_{\lim}$$
$$\text{for} \quad 1 \le i, j \le N_p, \text{ and } \; i \ne j \qquad (2)$$

where all combinations of i and j are considered. $\hat{n}_i$ and $\hat{n}_j$ are the normal vectors to body panels i and j, respectively, and N_p is the total number of body panels within a cell.

The methods described above will be applied to a Cartesian grid with the problems of thin body, sharp edge and two solid regions.

3. Flow Solver

The Euler equations are solved to calculate inviscid, compressible flows on a Cartesian grid generated by the present method. The flow solver employed here is based on the cell-centered finite-volume scheme, where the numerical flux is calculated using Hännel's flux-vector-splitting scheme.[12] The solution is advanced in time using a three-stage Runge-Kutta Method. Furthermore, convergence is accelerated using the local time stepping.

To achieve a reasonable time step, an extremely small and irregular cell produced by intersection of Cartesian grid with body surface is merged with its neighboring cell of larger size, which is carried out by taking the summation of mass, momentum and energy at all boundaries of the two cells. The same procedure is also used to merge the flow regions inside the cell for the case of two solid regions (see section 2.1.3).

4. Computation on Supersonic Transport Model

The present method was applied to a model of supersonic transport shown in Fig. 7a. The body surface consists of 9,906 triangular panels and 5,068 nodes for a semi-span model. Since the surface mesh of the model used here originally was not intended for CFD analysis, those triangles are of irregular size and shape. Such surface is not appropriate for body-fitted grid, whether structured or unstructured, because the smoothness of surface mesh is very important. Cartesian grid, on the other hand, is much less dependent on surface mesh quality. The only parameters that a user needs to set are the dimension of computational domain, the size of coarsest grid (level 0), and the highest level of grid refinement. The rest of the procedures are automatic.

Two tests are performed to examine: (1) the efficiency of the cell-splitting method, and (2) the effect of local grid refinement on accuracy. The flow is calculated under the conditions of M=2.4 and α=1.5°.

4.1 Cell-Splitting

An identical grid is used here, as shown in Figs. 7a and 7b. The grid is refined up to level 8, which corresponds to 538 cells along the axial length of aircraft. In other words, if the actual length of the aircraft is in the order of 100 m, the side of the smallest cubic cell will be about 19 cm. The cell geometrical properties are computed using two methods: the non-cell-splitting method as the baseline, and the cell-splitting method.

Pressure distributions on wing are shown in Figs. 9a and 9b. The pressure coefficient is plotted at 4 stations in the spanwise direction: η=0.3, 0.5, 0.7 and 0.9, where η is the ratio of the spanwise length to the half-span. When using the non-cell-splitting method, the thin-geometry problem is clearly seen even at η=0.3, especially near the trailing edge (see Fig. 9a).

By employing the cell-splitting method, a much more improved solution is obtained over the whole wing (see Fig. 9b). The flow solution near the leading edge, however, is less satisfactory, which is mainly due to low grid resolution in this region.

The total number of cells in the cell-splitting method is 265,418, which is a 1.3% increase compared with that of the non-cell-splitting method, which is not significant (see Fig. 8). On the other hand, the pseudo-plane approximation used in the cell-splitting method reduces the total number of body elements from 171,104 to 53,039, which is a factor of 3.2. As a result, the storage of the area and the normal vector of the panels decrease from 5.48MB in double precision for the baseline method to 1.70MB.

Although this reduces the total number of flux surfaces, which are composed of cell surfaces and body elements from 793,797 to 681,268, this advantage does not lead to a significantly faster flow computation, since there are other opposing factors that have greater effect on the speed of computation. The main factors are small size and shape irregularity of cells at body surface, which lowers CFL number, leading to slow computation. The typical flow computation time for the present configuration is in the order of one or two days on a NEC's EWS 4800/360MP workstation with R4400/100MHz processor.

The time to generate the grid using the current method is 1 hour 24 minutes of CPU time, where 67.6% of the total time is used for computing cell geometrical properties. 62% of this time span is used for checking *each* body panels for intersection with a particular cell at body surface. Since there are 53,039 cells at body surface and 9,906 original body panels, the checking algorithm is invoked 525 million times. A better approach to check only the panels in the vicinity of a particular cell is desired for further improvement.

4.2 Local Grid Refinement

A grid with local refinement around curved body surface is generated, so that the refinement level varies from 7 to 9 along the surface, which corresponds to cell sizes of about 37 cm to about 9 cm for SST of axial length 100 m (see Fig. 7c).

The grid contains 175,810 cells, which is a reduction of 34% over the cell-splitting method without local grid refinement (see Fig. 8). 18.6% of the total cells intersect with body surface. Regarding the original body surface elements, 87.2% of these cells have body panels with a connecting angle θ between 150° and 180°, which are considered to be nearly flat. On the other hand, the cells including a sharp edge with 0° ≤ $\theta \leq$ 30° account for 3%, all of which were split by the sharp-edge splitting method. Cells with 30° ≤ $\theta \leq$ 150° account for 9.8%, which are most refined. In particular, cells with 80° ≤ $\theta \leq$ 100°, which are considered to be least accurate due to the pseudo-planar approximation, account for 4.8%. It is expected that the present grid can capture the flow properties efficiently and effectively, because those unfavorable cells account for a small percentage and their sizes are the most refined.

It is evident that the pressure rises sharply near the leading edge, whereas the pressure distribution along the rest of the wing remains almost the same (see Figs. 9b and 9c). The number of body elements is also reduced by 61% to 32,616, compared with the case without local grid refinement. The storage for the area and the normal vector of panels takes about 1 MB, which shows that the cells are distributed much more efficiently.

However, the grid resolution around the leading edge is not yet sufficient even with the current refinement level. Here the average cell size is about 9 cm, whereas the curvature radius of the edge is in the order of 1 cm. Thus further refinement in this region is still necessary for more realistic applications.

5　Concluding Remarks

A cell splitting method for Cartesian grid was proposed in this study to deal with multiple flow or multiple solid regions within a cell, which occurs in the thin part of body. One of the features of the present method is to use pseudo-planar approximation for body elements within a cell, which can remarkably reduce computational cost. The method was further improved by considering the case of sharp edge as well as the case of two solid regions. Moreover, a local grid refinement method was also introduced at curved surface for efficient use of grid.

The present method was applied to a model of SST. The flow solution in a thin geometry region was improved by using only about 1% additional cells. Moreover, the number of body elements intersecting with grid cells was reduced by a factor of 3.2, due to the pseudo-planar approximation. Finally, the local grid refinement around curved surface further improved the flow solution using even fewer cells.

Acknowledgements

The authors gratefully acknowledge that the SST surface geometry used in this study was provided by Kawasaki Heavy Industries, Ltd.

References

1) Epstein, B, Luntz, A.L. and Nachshon, A.: Cartesian Euler method for Arbitrary Aircraft Configurations, AIAA Journal, vol. 30, no. 3, (1992), pp. 679-687.

2) De Zeeuw, D. and Powell, K.G.: An Adaptively Refined Cartesian Mesh Solver for the Euler Equations, J. of Comput. Physics, 104, (1993), pp. 56-68.

3) Pember, R.B., Bell, J.B., Colella, P., Crutchfield, W.Y. and Welcome, M.L.: An Adaptive Cartesian Grid Method for Unsteady Compressible Flow in Irregular Regions, J. of Comput. Physics, 120, (1995), pp. 278-304.

4) Ochi, A.: Numerical Calculation of SST using Unstructured Grid (in Japanese), Master thesis, Dept. of Aerospace Eng., Nagoya Univ., Japan, (1996).

5) Finley, D.B.: Euler Technology Assessment Program for Preliminary Aircraft Design Employing SPLITFLOW Code With Cartesian Unstructured Grid Method, NASA CR-4649, (1995).

6) Karman Jr., S.L.: Unstructured Cartesian / Prismatic Grid Generation for Complex Geometries, NASA CP-3291, (1995), pp. 251-270.

7) Melton, J.E., Berger, M.J., Aftosmis, M.J. and Wong, M.D.: 3D Applications of A Cartesian Grid Euler Method, AIAA-95-0853, (1995).

8) Lahur, P.R. and Nakamura, Y.: A New Method for Thin Body Problem in Cartesian Grid Generation, AIAA-99-0919, (1999).

9) Aftosmis, M.J. : Solution Adaptive Cartesian Grid Methods for Aerodynamic Flows with Complex Geometries, Computational Fluid Dynamics VKI Lectures Series 1997-05 (1997).

10) Aftosmis, M.J., Berger, M.J., and Melton, J.E.: Robust and Efficient Cartesian Mesh Generation for Component-Based Geometry, AIAA Journal, vol. 36, no. 6, (1998).

11) Deister, F., Rocher, D., Hirschel, E.H., and Monnoyer, F.: Three-Dimensional Adaptively Refined Cartesian Grid Generation and Euler Flow Solutions for Arbitrary Geometries, 4th European CFD Conference, Greece, vol. 1, part 1, (1998), pp. 96-101.

12) Hännel, D., Schwane, R., and Seider, G.: On the Accuracy of Upwind Schemes for the Solution of the Navier-Stokes Equations, AIAA Paper 87-1105, Proc. AIAA 8th Computational Fluid Dynamics Conference, pp. 42-46 (1987).

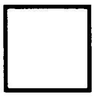
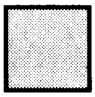
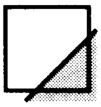
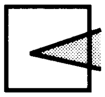
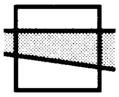
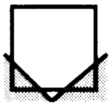
Class	Schematic view of cell
1	 fluid  solid
2	 normal cut  sharp edge
3	 thin body  2 solid regions
4	too complex to consider here

Fig. 1 Cases of intersection between grid cell and body surface according to the number of regions (fluid and solid) inside cell.

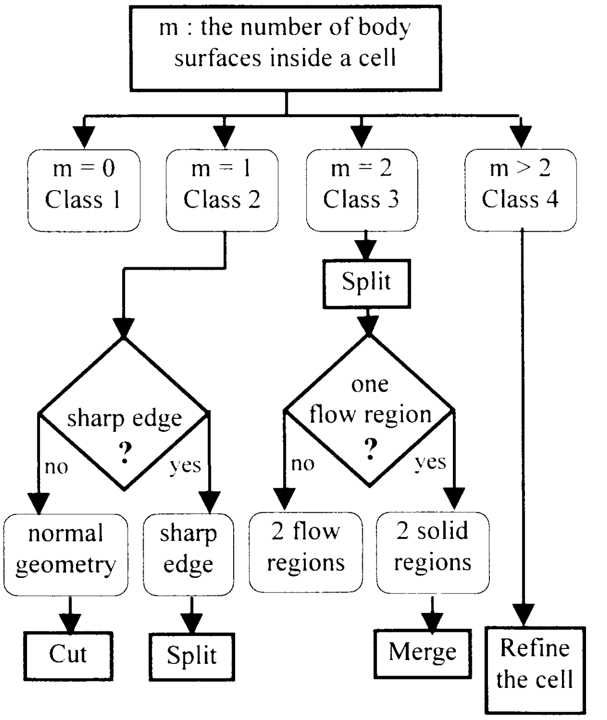


Fig. 2 Calculation procedure of cell geometrical properties.

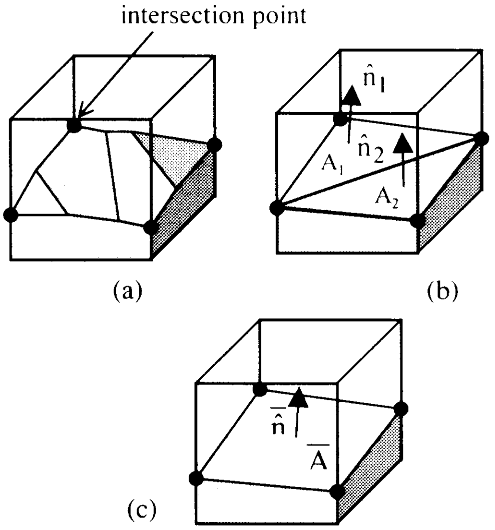


Fig. 3 Approximating body surface within a cell: (a) original body elements, (b) intermediate triangles, (c) pseudo-plane.

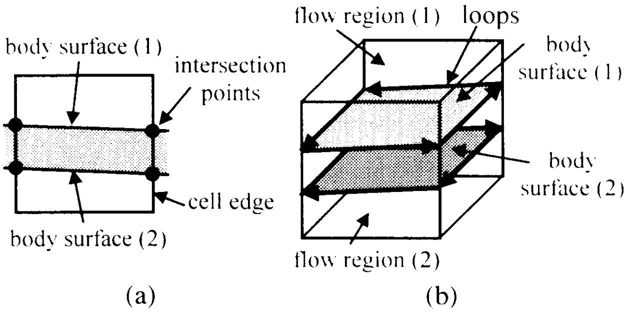


Fig. 4 Cell splitting procedure: (a) intersection between body surface and edges of a cell face, (b) construction of body surfaces inside cell.

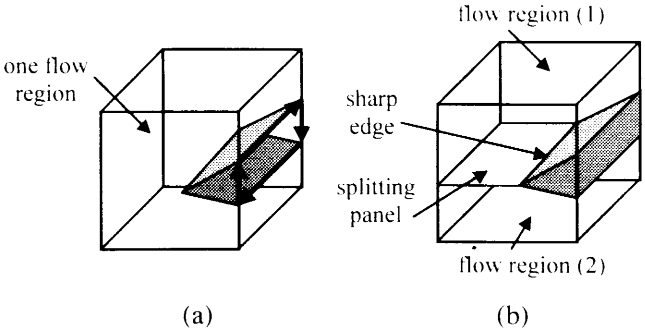


Fig. 5 Sharp edge treatment: (a) construction of *one* body surface, (b) splitting the cell.

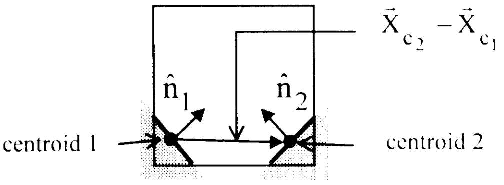


Fig. 6 Two solid regions in a cell.

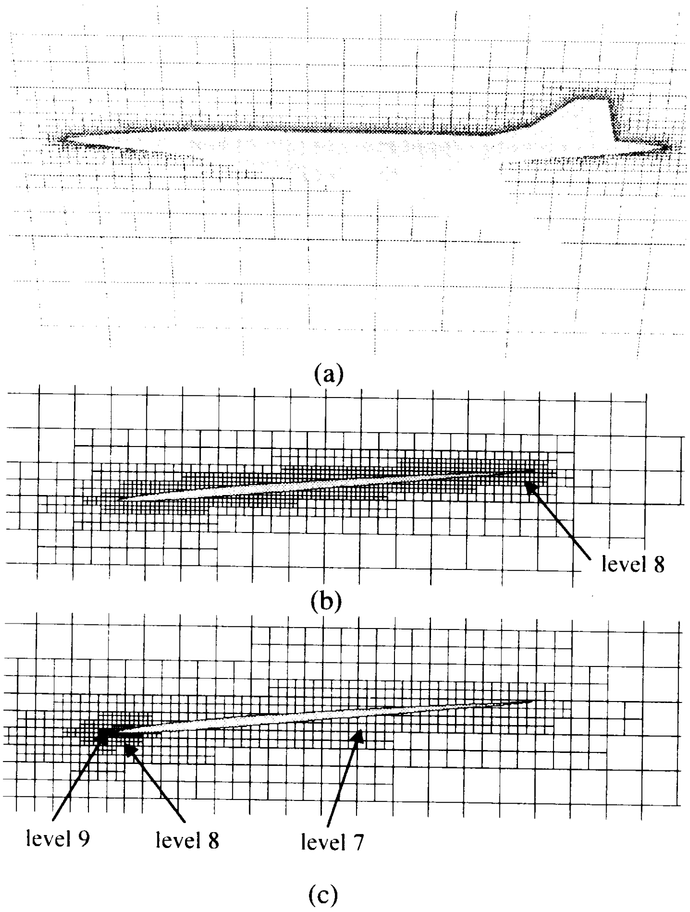


Fig. 7 Grid around SST:
(a) overall view, (b) cross-sectional view at η=0.5, (c) cross-sectional view at η=0.5 with local grid refinement near leading edge.

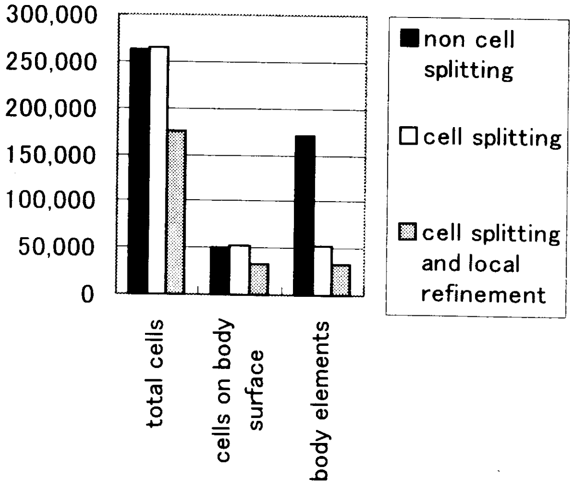


Fig. 8 Computational costs in terms of grid cells and body surface elements.

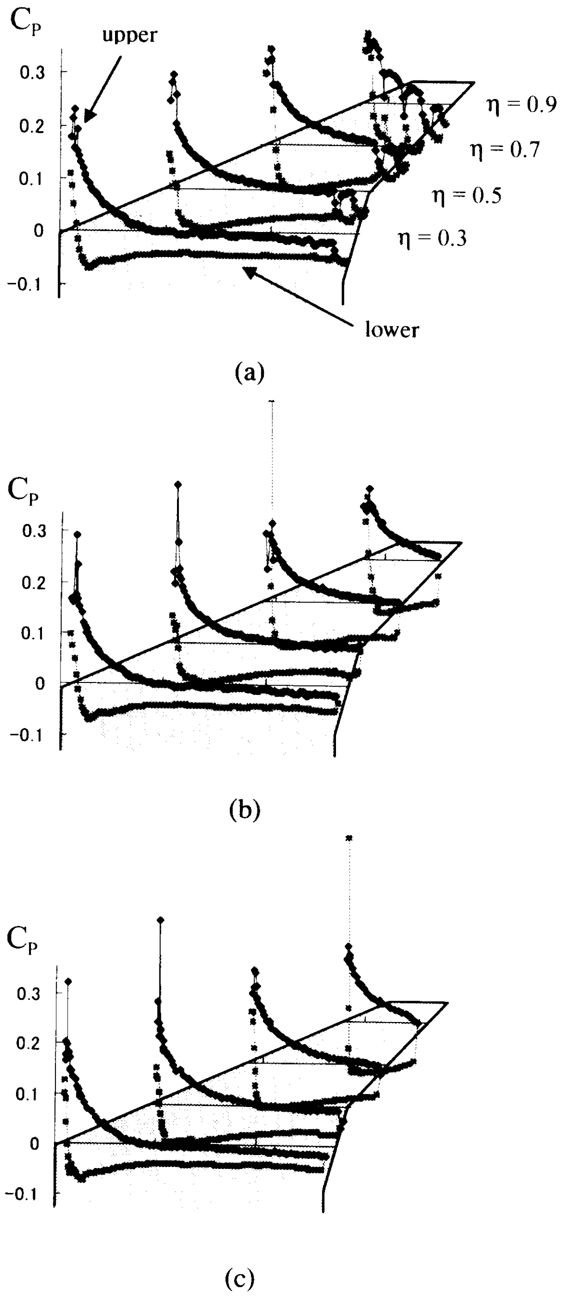


Fig. 9 Pressure distribution along SST wing:
(a) without cell splitting, (b) with cell splitting, (c) with cell splitting and local grid refinement.