AIAA JOURNAL
Vol. 45, No. 5, May 2007

Technical Notes

TECHNICAL NOTES are short manuscripts describing new developments or important results of a preliminary nature. These Notes should not exceed 2500 words (where a figure or table counts as 200 words). Following informal review by the Editors, they may be published within a few months of the date of receipt. Style requirements are the same as for regular contributions (see inside back cover).

Surface Mesh Movement for Aerodynamic Design of Body-Installation Junction

Hyoung-Jin Kim* and Kazuhiro Nakahashi†
Tohoku University, Sendai 980-8579, Japan

DOI: 10.2514/1.12947

I. Introduction

WITH the advances in computational fluid dynamics (CFD) and computing power of modern computers, aerodynamic shape design using CFD techniques has become more important than ever. In aerodynamic design of complex aircraft configurations with body-installation junctions such as wing-fuselage or nacelle-pylon-fuselage junctions, perturbation or modification of wing, fuselage, or nacelle-pylon geometries requires redefinition or modification of the junction lines and regeneration or movement of surface mesh points on the geometries.

For automated aerodynamic shape design of the body-installation junction regions, overset-grid methods can be applied with relative ease [1,2]. Another approach is a direct link of a CFD code and a CAD system [3], which requires regeneration of surface and volume meshes for every new design geometry.

As an alternative way without CAD intervention, Murayama et al. [4] developed a surface mesh movement method employing a surface mapping method in which a 3-D surface mesh region to be moved is partially cut out to make a patch and mapped onto a 2-D parameter domain. Meshes on the parameter domain are then modified appropriately and transformed back to the physical domain. For practical design problems, such parameterization is costly and causes inefficiency for the whole design process.

In this study, a new method for surface mesh movement is presented for practical aerodynamic shape design of body-installation junctions. Instead of the costly parameterization, we move the surface mesh points directly on the surface by using a spring analogy based on geodesic distances on the surface mesh. Time-consuming remeshing processes are not required.

As for the geometric representation, we only use the initial surface mesh, on which we search for new junction points and move the mesh points along. The new mesh points are projected onto the initial smooth surface by a surface recovery method. It also does not depend on design parameterization for geometry modification.

Received 24 August 2004; revision received 11 May 2006; accepted for publication 11 October 2006.

*Research Associate, Department of Aerospace Engineering, Aoba-yama 01; hjkim@ad.mech.tohoku.ac.jp. Member AIAA.

†Professor, Department of Aerospace Engineering, Aoba-yama 01; naka@ad.mech.tohoku.ac.jp. Associate Fellow AIAA.

II. Methodology

A. General Procedure

Figure 1 shows a schematic diagram for a wing-body configuration in which both wing and fuselage geometries are modified. Solid lines represent the initial geometry and dotted lines the modified geometry. Taking the wing-fuselage configuration as an example, the proposed procedure can be briefly described as follows:

1) Modify geometries of wing and fuselage.

2) For the modified geometries, determine a new location of wing-fuselage junction nodes.

3) Calculate geodesic distances from initial to new junction nodes along the surface mesh.

4) With the wing-fuselage junction node displacements as boundary conditions, move the surface mesh points on the initial fuselage facet surfaces using a spring analogy based on geodesic distances.

5) Do the same as steps 3 and 4 for mesh points on the wing facet surfaces.

6) The new surface mesh points on the initial faceted surfaces are projected onto the original surface by a surface recovery method.

7) Modify the initial volume mesh according to the movement of surface mesh points using a 3-D volume mesh modification method.

B. Finding New Junction Node Positions

When the undefined wing geometry inside the fuselage is exposed by the geometric modification, the wing surface geometry needs to be extrapolated spanwisely to find the new junction node positions. For structured grids, spanwise grid lines on wing surfaces are usually along the local sweep angles of the wing. Thus, geometric extrapolation in the spanwise direction can be easily done along the grid lines [5,6].

Because no aligned grid lines exist in unstructured meshes, we calculate an extension vector $\boldsymbol{T}$, which is along a local sweep angle and tangential to a wing surface cell at every junction node as shown in Fig. 2. The local sweep angle is calculated by a linear interpolation between leading- and trailing-edge sweep angles. A tangential extension vector $\boldsymbol{T}$ is then calculated preserving the local sweep angle:

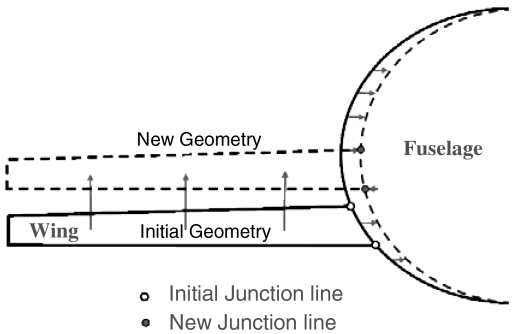


Fig. 1 Schematic diagram for geometry modification with wing-body junction.

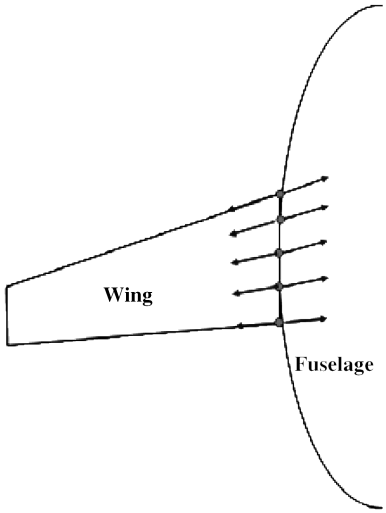


Fig. 2 Spanwise extrapolation vector for sweep angle preservation.

$$\boldsymbol{T} = \boldsymbol{S} - (\boldsymbol{S}\cdot\boldsymbol{n})\boldsymbol{n}, \qquad \boldsymbol{S} = (\sin\Lambda_{\text{local}}, \cos\Lambda_{\text{local}}, 0) \tag{1}$$

where Λ_{local} is the local sweep angle, $\boldsymbol{n}$ the normal vector of a facet cell on the wing surface. The z component of $\boldsymbol{T}$ tends to contain some oscillations, which need to be removed by a smoothing technique.

Although this simple wing extension might look somewhat crude from a geometric viewpoint, it is very effective for aerodynamic configurations such as wings and pylons having lateral extensions by nature. Furthermore it is robust for the small amount of geometric variation usually encountered in aerodynamic shape design of the preliminary design phase.

The next step is to find a new junction point where the extrapolation vector meets the fuselage surface mesh. This is done by using a neighbor-to-neighbor jump search method (see Fig. 3). Although the neighbor-to-neighbor search enables an efficient 1-D search, it may fail to find the intersection cell if there is a hole between the initial and intersection node points. This is the case for node j on the lower junction line in Fig. 3. To circumvent this problem, subsidiary cells are added in the junction hole as shown by dotted lines. The subsidiary cells are also helpful for efficient interpolation of the undefined fuselage geometry on the hole by a surface recovery method.

C. Geodesic Distance

When the new junction point is found by the neighbor-to-neighbor search method, one needs to calculate geodesic distances between the initial and new junction points, because these are used as boundary conditions for surface mesh movement. The geodesic distance is the distance between points along curved surfaces. The Euclidian distance approximates the geodesic distance well only at short ranges and regions of small curvature. Although the Euclidian distance is much simpler to compute, the geodesic distance would give more robust results because the distance traveled by the boundary junction node is the geodesic distance, which is always larger than the Euclidian distance. If the Euclidian distance is used for the surface mesh movement problem, flipped surface meshes may result due to the junction movement being greater than the inner surface nodes.

Geodesics preserve the direction on a surface, and the normal vector to any point of a geodesic arc lies along the direction normal to a surface at that point [7]. Considering these properties, the geodesic distance can be calculated approximately on the facets by the following procedure (see Fig. 3):

1) Set node i (initial junction point) as a starting point and node i' (new junction point) as a target point. Set a geodesic distance $G = 0$.

2) Select a facet cell (cell a in Fig. 3, for instance) having node i as one of its three vertices and lying on the way from the starting point to the target point.

3) Calculate a direction vector $\boldsymbol{V}$ from the starting point to the target point.

4) Project the direction vector $\boldsymbol{V}$ to the cell plane and keep the vector length unchanged as follows:

$$\boldsymbol{R} = |\boldsymbol{V}| \cdot \frac{\boldsymbol{V} - (\boldsymbol{V}\cdot\boldsymbol{n})\boldsymbol{n}}{|\boldsymbol{V} - (\boldsymbol{V}\cdot\boldsymbol{n})\boldsymbol{n}|} \tag{2}$$

where $\boldsymbol{R}$ is the resultant vector, and $\boldsymbol{n}$ is the normal vector of the current surface cell.

5) Find an intersection between the vector $\boldsymbol{R}$ and three edges of the cell.

6) If $\boldsymbol{R}$ and the edges of the current cell intersect somewhere besides the starting point, calculate the distance from the starting point to the intersection point, and add it to the geodesic distance G. Then move to the cell located on the other side of the edge on which the intersection point lies. Set the intersection point as a new starting point and go on to step 3.

7) If $\boldsymbol{R}$ and the edges do not intersect except at the starting point, the target point is inside the current cell. In this case, calculate the distance from the starting point to the target point and add it to the geodesic distance G. Go on to step 9.

8) Otherwise repeat steps 1–7 for all junction points.

9) End of the procedure.

Step 4 for direction vector rotation is schematically presented in Fig. 4 for a simplified section view. The calculation of intersection points between vector $\boldsymbol{R}$ and cell edges renders an overdetermined matrix system, which can be solved by the singular value decomposition method [8].

D. Spring Analogy

With the junction node displacements as boundary conditions, surface mesh points are moved using a spring analogy. Spring force on node i by edge connecting nodes i and j can be represented as follows:

$$\boldsymbol{F}_{ij} = k_{ij}\Delta\boldsymbol{d}_{ij}, \qquad \Delta\boldsymbol{d}_{ij} = \boldsymbol{d}_i - \boldsymbol{d}_j \tag{3}$$

where the movement vector $\boldsymbol{d}$ is defined by the direction vector and the geodesic distance between the initial and final points as follows:

$$\boldsymbol{d} = GV/|\boldsymbol{V}| \tag{4}$$

where G is the geodesic distance.

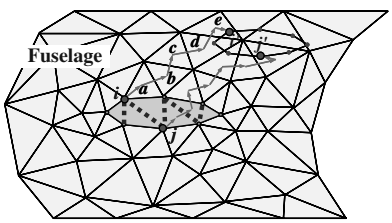
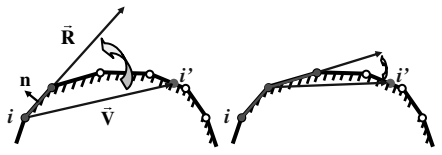


Fig. 3 Neighbor-to-neighbor search from initial to new junction location (subsidiary cells are created by adding the edges indicated by dotted lines).



a) Procedure for the first cell **b) Procedure for the second cell**

Fig. 4 Schematic diagrams for calculation of the geodesic distance from i to i'.

The stiffness coefficient of edge spring k_{ij} is defined as inversely proportional to edge length and distance from the junction line:

$$k_{ij} = \frac{1}{l_{ij}(n\text{dist}_j + 1)^2}$$

where l_{ij} is the edge length and $n\text{dist}_j$ is the minimum number of edges from the junction nodes to node j. The required movement vector $\boldsymbol{d}$ can be obtained by the force equilibrium at each node point. The linear system of equations for the spring analogy given in Eq. (3) is solved by the point Gauss–Seidel iteration method.

The surface mesh points are then moved along $\boldsymbol{d}$ on the surface facets. This procedure is similar to the procedure for the geodesic distance calculation for a junction point in Sec. II.C, except that now the movement vector $\boldsymbol{d}$ with a norm of the geodesic distance is given instead of the target point. Vector $\boldsymbol{d}$ is rotated so as to be tangential to the cell and the traveled distance is summed up until it reaches the given geodesic distance for the node.

E. Surface Recovery

New mesh points are now located on facets of the initial background surface mesh. The new nodes need to be projected onto the smooth surface geometry. This can be achieved by either a direct CAD interface method [9] or a surface recovery method [10]. In the present study, we adopt the latter approach for a more portable and simple design/analysis tool.

The recovered point location is usually given by a quadratic triangle shape function. The midpoint location $\boldsymbol{x}_M$ is estimated by a Hermitian polynomial as

$$\boldsymbol{x}_M = 0.5(\boldsymbol{x}_1 + \boldsymbol{x}_2) + 0.125(\boldsymbol{r}_1 - \boldsymbol{r}_2) \tag{5}$$

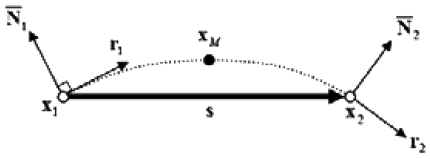
where

$$\boldsymbol{r}_i = |\boldsymbol{s}| \cdot \frac{\bar{\boldsymbol{N}}_i \times (\boldsymbol{s} \times \bar{\boldsymbol{N}}_i)}{|\bar{\boldsymbol{N}}_i \times (\boldsymbol{s} \times \bar{\boldsymbol{N}}_i)|}, \qquad (i = 1, 2), \qquad \boldsymbol{s} = \boldsymbol{x}_2 - \boldsymbol{x}_1$$

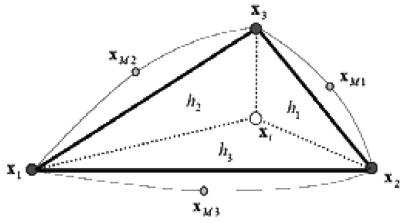
$\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are location vectors of both ends of the edge; $\bar{\boldsymbol{N}}_1$ and $\bar{\boldsymbol{N}}_2$ are unit normal vectors at each node (see Fig. 5a). Note that these nodes and edges belong to the background mesh. If node i of the new surface mesh is on a triangular facet of the background mesh having three nodes $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$, the recovered node location is

$$\boldsymbol{x}_i' = h_1(2h_1 - 1)\boldsymbol{x}_1 + h_2(2h_2 - 1)\boldsymbol{x}_2 + h_3(2h_3 - 1)\boldsymbol{x}_3 + 4(h_2h_3\boldsymbol{x}_{M1} + h_3h_1\boldsymbol{x}_{M2} + h_1h_2\boldsymbol{x}_{M3})$$

where h_1, h_2, and h_3 represent the area coordinate of node I and $\boldsymbol{x}_1$, $\boldsymbol{x}_2$, and $\boldsymbol{x}_3$ represent the midpoint location vectors of each edge as shown in Fig. 5b.



a) Midpoint location vector $\mathbf{x}_M$



b) A facet of the background mesh and node i of the new surface mesh

Fig. 5 Quadratic surface recovery.

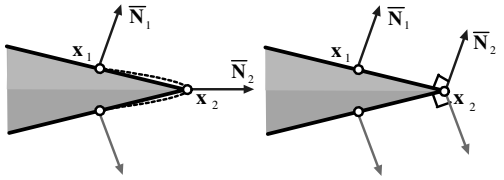


Fig. 6 Definition of normal vectors on ridges such as the trailing edge for surface recovery: single normal vector (left) and multiple normal vectors (right). Single normal vector is not adequate for trailing edges.

If $\boldsymbol{x}_1$ or $\boldsymbol{x}_2$ is on a ridge such as wing trailing edge, the midpoint location may deviate from the initial geometry as indicated by the dotted line in Fig. 6. To circumvent this problem, multiple normal vectors are defined at ridge nodes when the angle between normal vectors of two adjacent surface cells is larger than 60 deg as shown in Fig. 6 [11].

III. Results

For the validation of the present method, ONERA M5 wing-body and DLR-F6 wing-body-nacelle-pylon (WBNP) configurations, including their junction regions, are modified.

A. ONERA M5 Configuration

A surface mesh of the ONERA M5 wing-body configuration is shown in Fig. 7. Figure 8 represents fuselage surface meshes and subsidiary cells on the junction hole. As a first example for ONERA M5, the main wing is moved vertically for a considerable distance. Surface mesh points both on the wing and fuselage surfaces are moved accordingly following the procedure described in the preceding sections. As can be seen in Fig. 9 the fuselage and wing surface meshes are successfully moved according to the vertical movement of main wing, although the tested amount of movement is much larger than usual. The main wing surface meshes before and after the modification are presented in Fig. 10, which shows that the mesh movement was properly conducted. However, if mesh quality after a considerable geometry modification is important, local remeshing can be easily combined with the present method.

As a second example, the fuselage is modified so as to have a narrower width, which results in exposure of the undefined wing geometry. Figure 11 shows that such an undefined region is

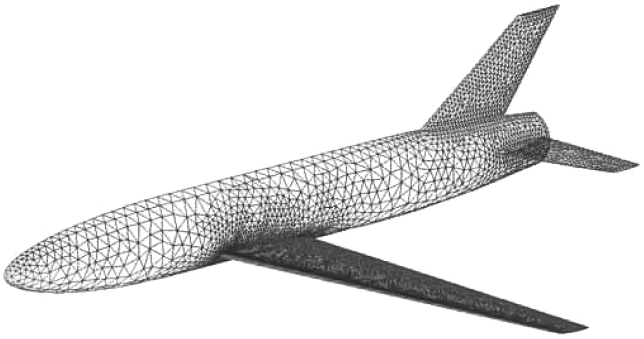


Fig. 7 Surface mesh for ONERA M5 wing.

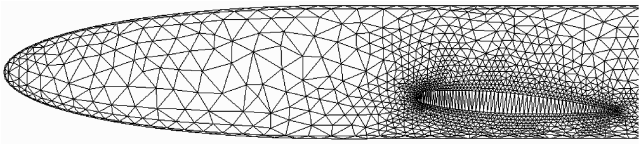


Fig. 8 Side view of ONERA M5 fuselage surface mesh with subsidiary cells on the junction hole.

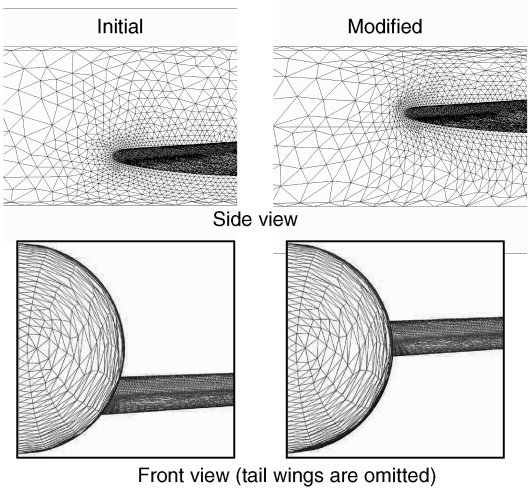


Fig. 9 Vertical movement of ONERA M5 main wing (left: initial; right: modified).

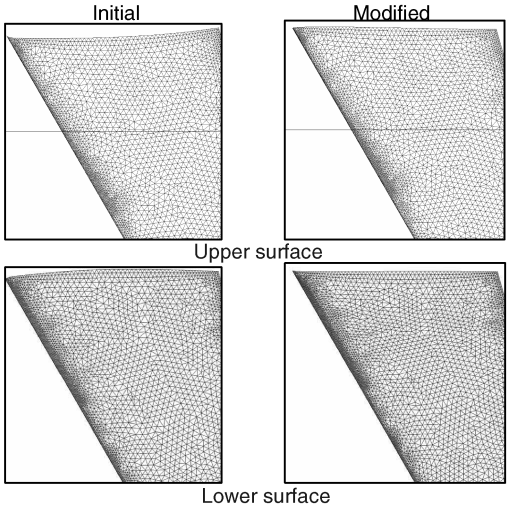


Fig. 10 Wing surface mesh for vertical movement of ONERA M5 main wing (left: initial; right: modified).

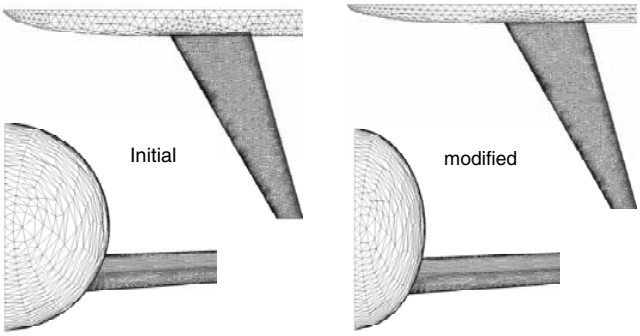


Fig. 11 ONERA M5 fuselage modification.

successfully extrapolated by the extension vectors and that surface meshes are stretched to fill the newly exposed region.

B. DLR-F6 WBNP Configuration

The next example is the DLR-F6 wing-body-nacelle-pylon (WBNP) configuration (see Fig. 12). In this case, we consider translational movements of the nacelle assuming that the pylon-nacelle junction is not altered and that the pylon moves according to the movement of the nacelle.

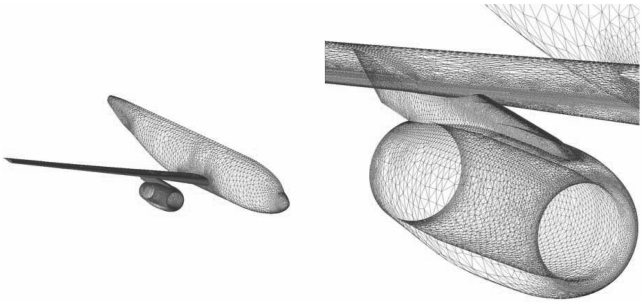


Fig. 12 DLF-F6 WBNP configuration and surface mesh.

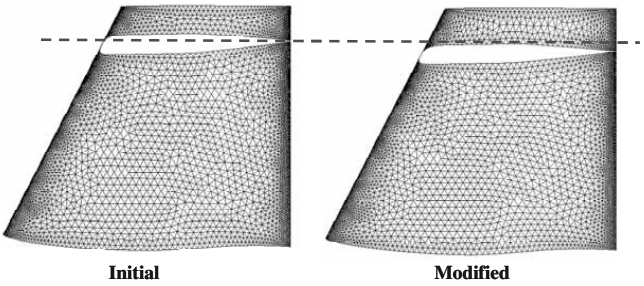


Fig. 13 Inboard lower surface mesh before/after spanwise nacelle movement ($\Delta y = -0.005$).

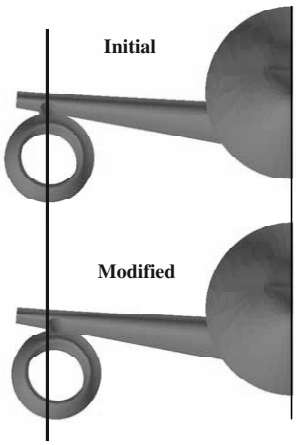


Fig. 14 Front view of nacelle movement in spanwise direction with pylon deformation.

As can be seen in Fig. 12, the pylon is installed on the lower surface of the main wing. The pylon chord length is limited by the wing section chord length, which means that pylon deformation is required for nacelle movement. For a given lateral displacement, pylon deformation is conducted by a simple spring analogy with the spring constant being based on edge length. The proposed procedure for the body-installation junction is then conducted.

Figure 13 shows inboard wing lower surface meshes before and after an inward nacelle movement of $\Delta y = -0.005$. The pylon is stretched by the deformation because the inner wing sections have longer chord lengths. Figure 14 shows a front view of the DLR F6 model.

The computational time for the test case of DLR-F6 is about 100 s at Pentium-4 3.0 GHz, most of which is required for solution of the spring analogy for the surface mesh movement. This can be improved by using a numerical method which is more efficient than the point Gauss–Seidel method for the spring analogy.

IV. Conclusions

A surface mesh movement method was developed for practical modification of geometries with body-installation junctions. We moved the surface mesh points directly on the facet surfaces using a spring analogy. The initial mesh is used as the background mesh. New mesh locations are projected onto the initial smooth surface by a surface recovery method. This method does not require any parameterization or geometry representations other than the faceted surface mesh. Neither does it require time-consuming remeshing processes. It allows efficient and robust mesh movement for modified geometries of wing-body configurations including the junction region, which is essential for aerodynamic design of complex configurations such as an entire aircraft.

For mild modifications usually encountered in aerodynamic shape optimizations, this method is robust as shown by the examples herein presented. If the geometric variations are so large that simple surface mesh movement causes highly stretched cells, mesh point insertion/deletion and local remeshing for mesh quality control can additionally be conducted. The present method does not work if a large amount of geometric modification causes removal or topological change of the junction lines. The present procedure can also be applied to wing-body junctions with fairings as long as junction lines having gradient discontinuities can be defined.

In the present study, this new method was applied to ONERA M5 wing-body and DLR F6 wing-body-nacelle-pylon configurations with somewhat large geometric modifications. Results indicate that the present method is very efficient and effective for practical aerodynamic shape design of complex aerodynamic configurations with junctions.

Acknowledgment

The authors wish to thank Mitsubishi Heavy Industries, Ltd. for their financial support of this work.

References

[1] Steger, J. L., Dougherty, F. C., and Beneck, J. A., "A Chimera Grid Scheme," *Advances in Grid Generation*, edited by K. N. Ghia and U. Ghia, American Society of Mechanical Engineers, Fairfield, NJ, June 1983, pp. 59–69.

[2] Nakahashi, K., Togashi, F., and Sharov, D., "Integrated-Boundary Definition Method for Overset Unstructured Grid Approach," *AIAA Journal*, Vol. 38, No. 11, 2000, pp. 2077–2084.

[3] Jameson, A., Martinelli, L., and Haimes, B., "Aerodynamic Shape Optimization of Complete Aircraft Configurations Using Unstructured Grids," AIAA Paper 2004-0533, Jan. 2004.

[4] Murayama, M., Nakahashi, K., and Matsushima, K., "A Robust Method for Unstructured Volume/Surface Mesh Movement," *Transactions of the Japan Society for Aeronautical and Space Sciences*, Vol. 46, No. 152, Aug. 2003, pp. 104–112.

[5] Barger, R. L., and Adams, M. S., "Automatic Procedures for Computing Complete Configuration Geometry from Individual Component Descriptions," NASA Technical Memorandum 4607, July 1994.

[6] Sasaki, D., Yang, G., and Obayashi, S., "Automated Aerodynamic Optimization Systems for SST Wing-Body Configurations," *Transactions of the Japan Society for Aeronautical and Space Sciences*, Vol. 46, No. 154, 2004, pp. 230–237.

[7] Weinstock, R., *Calculus of Variations, with Application to Physical Engineering*, Dover, New York, 1974, pp. 26–28, 45–46.

[8] Press, W. H., Teukosky, S. A., Vetterling, W. T., and Flannery, B. P., *Numerical Recipes in Fortran*, 2nd ed., Cambridge Univ. Press, Cambridge, England, 1992, pp. 51–63.

[9] Park, M. A., "Adjoint-Based, Three-Dimensional Error Prediction and Grid Adaptation," *AIAA Journal*, Vol. 42, No. 9, 2004, pp. 1854–1862.

[10] Löhner, R., "Regridding Surface Triangulations," *Journal of Computational Physics*, Vol. 126, No. 1, 1996, pp. 1–10.

[11] Ito, Y., and Nakahashi, K., "Direct Surface Triangulation Using Stereolithography Data," *AIAA Journal*, Vol. 40, No. 3, 2002, pp. 490–496.

K. Ghia
Associate Editor