**JOURNAL OF THEORETICAL AND APPLIED MECHANICS**
**61**, 3, pp. 453-463, Warsaw 2023
https://doi.org/10.15632/jtam-pl/165954

# ROBUST GENERATION METHOD OF A SIGNED DISTANCE FUNCTION FOR PREPROCESSING OF CARTESIAN-GRID-BASED CFD

Yuki Takeda, Kazuyuki Ueno, Yuta Takahashi, Karin Matsubara
*Faculty of Science and Engineering, Iwate University, Japan*
*correspondin author Yuki Takeda, e-mail: takeday@iwate-u.ac.jp*

In practical computational fluid dynamics simulations around industrial products with complex surface shapes, the robustness of preprocessing to "dirty" geometry is an important issue. The dirty STL (Standard Triangle Language) data contains errors such as gaps between facets, overlapping of facets, and flipping of normal vectors. These errors in the STL data are difficult to avoid in 3D modeling of complex geometry. Using a Cartesian grid is advantageous to the boundary-fitted grid in the aspect of preprocessing for dirty STL files. In this study, a robust and automatic generation method of a signed distance function for the preprocessing of Cartesian grid solvers is proposed. To ensure robustness to the complex and dirty STL data, the proposed method uses information of all STL facets to determine each grid point. The proposed preprocessing method is verified by numerical simulation of the flow around the NASA common research model.

*Keywords:* Cartesian grid dolvers, computational fluid dynamics, signed distance function, preprocessing

## 1. Introduction

The Cartesian grid system enables easy and automatic generation of computational grids. A favorable aspect of the Cartesian grid is that it does not require the grid system to fit the shape of the object. The voxel method (Ishida *et al.*, 2008), immersed boundary method (Peskin, 1977; Iaccarino and Verzicco, 2003) and Cartesian cut-cell method (Hu *et al.*, 2006; Takeda *et al.*, 2020a,b) are typical Cartesian grid solvers in computational fluid dynamics (CFD).

Cartesian grid solvers require an algorithm to detect the object surface to predict the flow along the surface in body-unfitted grid systems. A scalar field function that defines the object shape is commonly used in the surface detection algorithm. A signed distance function is a widely used indicator to define the object shape in Cartesian grid solvers. In the case of a flow around a rigid and static body, the signed distance function is generated in the preprocessing, and it is not updated during the main process. The level-set function in the level-set method (Sussman *et al.*, 1994; Estellers *et al.*, 2012) is a famous application of the signed distance function in CFD.

The absolute value of the signed distance function is equivalent to the distance function from the object surface. The signed distance function has a positive value when the point is outside the object and a negative value when the point is within the object. This function around a simple object shape is easily generated by a combination of piecewise elementary scalar field functions. However, object shapes associated with the computer-aided design (CAD) in practical numerical simulations cannot be described by elementary functions.

The STL file format (Szilvási-Nagy and Mátyási, 2003) in CAD software is often used to generate the signed distance function around complex geometry. In STL files, the object shape is reproduced by triangular facets, and large curvature is represented by refinement of these facets (Szilvási-Nagy and Mátyási, 2003). The STL file comprises vertices of triangular facets and unit

normal vectors of the facets. The unit normal vector defines the front or back side of each facet. Usually, the facet normal vector is defined in the outward direction of the object. However, some so-called "dirty" STL files (Lahur *et al.*, 2012) contain errors such as gaps between facets, overlapping of facets, and flipping of normal vectors. These errors are caused by the treatment of the 3D object shape in CAD software.

In 3D-CAD software, it is difficult to avoid generating of this dirty STL data. The robustness of generation of the signed distance function to dirty geometry is effective in reducing the lead time of CFD. In this study, a robust and automatic generation method of the signed distance function for Cartesian grid-based CFD is proposed. The quality of the generated signed distance function is verified by the benchmark problem in CFD.

## 2. Numerical method for calculating the signed distance field

### 2.1. Calculation of the distance from a triangular facet

An arbitrary triangular facet that constitutes STL data of an object surface is defined as a triangle $ABC$, as shown in Fig. 1, and this section determines the distance to an arbitrary grid point $D$ in a 3D space. The volume $V$ of tetrahedron $ABCD$ is expressed using vectors $\overrightarrow{AB} = \mathbf{b}$, $\overrightarrow{AC} = \mathbf{c}$, and $\overrightarrow{AD} = \mathbf{d}$ as follows

$$V = \frac{1}{6}(\mathbf{b} \times \mathbf{c}) \cdot \mathbf{d} \tag{2.1}$$

The volume $V$ is also expressed as

$$V = \frac{1}{3} S_{ABC} |\overrightarrow{ED}| \tag{2.2}$$

where $S_{ABC} = |\mathbf{b} \times \mathbf{c}|/2$ is the area of triangle $ABC$ and $|\overrightarrow{ED}|$ is height of tetrahedron $ABCD$. From Eqs. (2.1) and (2.2), distance $d_E$ is expressed as follows

$$d_E = |\overrightarrow{ED}| = \frac{(\mathbf{b} \times \mathbf{c}) \cdot \mathbf{d}}{|\mathbf{b} \times \mathbf{c}|} \tag{2.3}$$

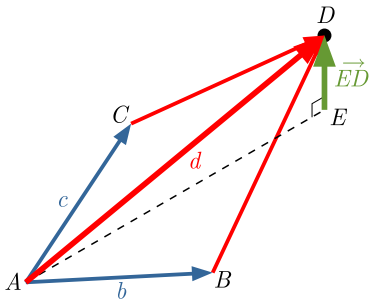


Fig. 1. Geometry of tetrahedron $ABCD$ and foot $E$

Let us introduce $d_{ABC}$ as the distance between point $D$ and the triangular facet. This distance is calculated according to the positional relationship between triangle $ABC$ and foot $E$ perpendicular to plane $D$, including triangle $ABC$. Expressing the vector $\overrightarrow{AE} = k_1\mathbf{b} + k_2\mathbf{c}$, the vector $\overrightarrow{AE}$ is

$$\overrightarrow{ED} = \mathbf{d} - (k_1\mathbf{b} + k_2\mathbf{c}) \tag{2.4}$$

The vector $\overrightarrow{ED}$ is orthogonal to the vectors $\mathbf{b}$ and $\mathbf{c}$

$$\mathbf{b}\cdot[\mathbf{d}-(k_1\mathbf{b}+k_2\mathbf{c})]=0 \qquad \mathbf{c}\cdot[\mathbf{d}-(k_1\mathbf{b}+k_2\mathbf{c})]=0 \tag{2.5}$$

Coefficients $k_1$ and $k_2$ are obtained by solving simultaneous Eq. (2.5)

$$\begin{bmatrix}k_1\\k_2\end{bmatrix}=\frac{1}{Det}\begin{bmatrix}\mathbf{c}\cdot\mathbf{c} & -\mathbf{b}\cdot\mathbf{c}\\ -\mathbf{b}\cdot\mathbf{c} & \mathbf{b}\cdot\mathbf{b}\end{bmatrix}\begin{bmatrix}\mathbf{b}\cdot\mathbf{d}\\ \mathbf{c}\cdot\mathbf{d}\end{bmatrix} \tag{2.6}$$
$$Det=(\mathbf{b}\cdot\mathbf{b})(\mathbf{c}\cdot\mathbf{c})-(\mathbf{b}\cdot\mathbf{c})^2$$

The positional relationships between triangle $ABC$ and point $E$ expressed by $k_1$ and $k_2$ are shown in Fig. 2. The distance $d_{ABC}$ is determined via different formulas according to the values of $k_1$ and $k_2$. In the case of point $E$ inside the triangle, the distance $d_{ABC}$ coincides with $d_E$

$$d_{ABC}=\frac{(\mathbf{b}\times\mathbf{c})\cdot\mathbf{d}}{|\mathbf{b}\times\mathbf{c}|} \tag{2.7}$$

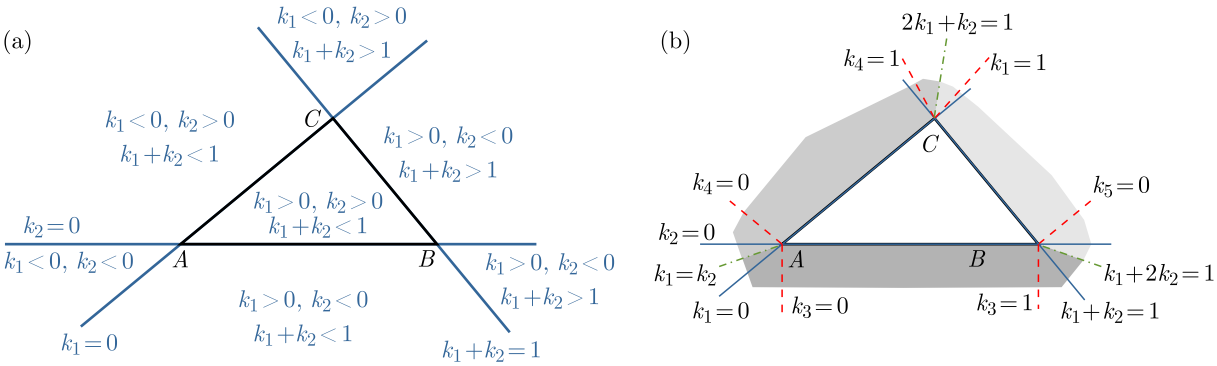


Fig. 2. Distribution of the functions $k_1$, $k_2$, $k_3$ and $k_4$ around triangle $ABC$: (a) distribution of the solutions $k_1$ and $k_2$, (b) relation between $(k_1, k_2)$ and the position of foot $E$ in the perpendicular direction

If point $E$ is outside the triangle, $d_{ABC}$ is determined by Table 1 using the area division in Fig. 2b. The coefficients $k_3$, $k_4$ and $k_5$ are as follows

$$k_3=\frac{\mathbf{b}\cdot\mathbf{d}}{\mathbf{b}\cdot\mathbf{b}} \qquad k_4=\frac{\mathbf{c}\cdot\mathbf{d}}{\mathbf{c}\cdot\mathbf{c}} \qquad k_5=\frac{(\mathbf{c}-\mathbf{b})\cdot(\mathbf{d}-\mathbf{b})}{(\mathbf{c}-\mathbf{b})\cdot(\mathbf{c}-\mathbf{b})} \tag{2.8}$$

**Table 1.** Area division of point $E$ placed outside the triangle $ABC$

| Position of point $E$ (range of $k_1$ and $k_2$) | Range of coefficients $k_i$ and value of $d_{ABC}$ | | |
|---|---|---|---|
| Vicinity of side $AB$ ($k_1 > k_2$ and $k_1 + 2k_2 < 1$) | $k_3 \leqslant 0$ $\lvert\mathbf{d}\rvert$ | $0 < k_3 < 1$ $\lvert\mathbf{b}\times\mathbf{d}\rvert/\lvert\mathbf{b}\rvert$ | $k_3 \geqslant 1$ $\lvert\mathbf{d}-\mathbf{b}\rvert$ |
| Vicinity of side $AC$ ($k_1 < k_2$ and $2k_1 + k_2 < 1$) | $k_4 \leqslant 0$ $\lvert\mathbf{d}\rvert$ | $0 < k_4 < 1$ $\lvert\mathbf{c}\times\mathbf{d}\rvert/\lvert\mathbf{c}\rvert$ | $k_4 \geqslant 1$ $\lvert\mathbf{d}-\mathbf{c}\rvert$ |
| Vicinity of side $BC$ ($k_1 + 2k_2 > 1$ and $2k_1 + k_2 > 1$) | $k_5 \leqslant 0$ $\lvert\mathbf{d}-\mathbf{b}\rvert$ | $0 < k_5 < 1$ $\lvert(\mathbf{c}-\mathbf{b})\times(\mathbf{d}-\mathbf{b})\rvert/\lvert\mathbf{c}-\mathbf{b}\rvert$ | $k_5 \geqslant 1$ $\lvert\mathbf{d}-\mathbf{c}\rvert$ |

The distance from the object surface is obtained by sweeping all triangular facets. The minimum value of $d_{ABC}$ is the absolute value $|\phi|$ of the signed distance function $\phi$. To determine $|\phi|$, only the values of the nearest facet for point $D$ are selected, and other facets have no effect after the selection.

### 2.2. Sign evaluation of the signed distance function $\phi$

The sign of the signed distance function is determined by considering the positional relationship with the object. This inside/outside evaluation regarding the object (represented by a polyhedron) is called the point-in-polyhedron test (Carvalho and Cavalcanti, 1995; Baerentzen and Aanaes, 2005; Li and Wang, 2017). The ray casting method (Roth, 1982) and the method using the sign of the inner product are well-known and simple implementations of the point-in--polyhedron testing. However, these implementations are non-robust against errors in the STL data file because they refer to limited number of facets to detect the sign of a grid point. This study implements a point-in-polyhedron testing method using summation of the solid angle. In contrast to the conventional method, this method uses the information of all STL facets to determine the sign of each grid point. This method refers to the point-in-polyhedron testing method originally proposed by Carvalho and Cavalcanti (1995) in the context of computer graphics.

The solid angle $\omega$ is defined as shown in Fig. 3 as the area of a spherical triangle on a unit sphere centered at grid point $D$. The absolute value $|\omega|$ of the solid angle $\omega$ is given by

$$|\omega| = \alpha + \beta + \gamma - \pi \tag{2.9}$$

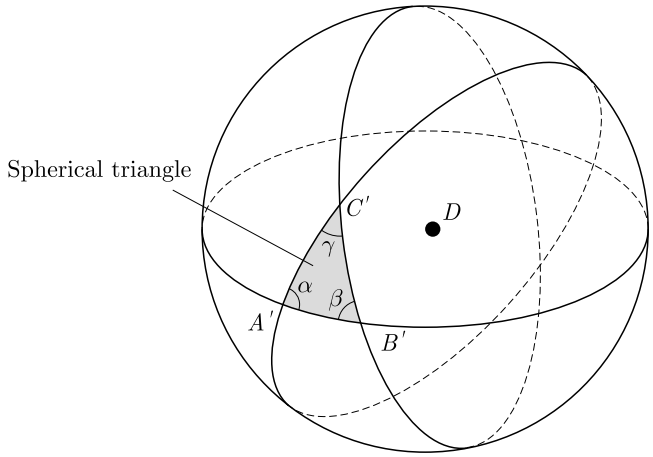


Fig. 3. Definition of the spherical triangle $A'B'C'$ on a unit sphere

We consider the interior angles $\alpha$, $\beta$ and $\gamma$. The intersection of the straight line $DA$ and the unit sphere centered at $D$ is defined as point $A'$. Points $B'$ and $C'$ are defined in the same way

$$\mathbf{a}' = \overrightarrow{DA'} = -\frac{\mathbf{d}}{|\mathbf{d}|} \qquad \mathbf{b}' = \overrightarrow{DB'} = \frac{\mathbf{b}-\mathbf{d}}{|\mathbf{b}-\mathbf{d}|} \qquad \mathbf{c}' = \overrightarrow{DC'} = \frac{\mathbf{c}-\mathbf{d}}{|\mathbf{c}-\mathbf{d}|} \tag{2.10}$$

The normal vector of the plane that includes the great circle $A'B'$ shown in Fig. 3 is expressed by $\overrightarrow{DA'} \times \overrightarrow{DB'}$, while the normal vector of the plane that includes the great circle $A'C'$ is $\overrightarrow{DA'} \times \overrightarrow{DC'}$. The angle between these two planes is equivalent to the interior angle $\alpha$

$$\alpha = \arccos \frac{(\mathbf{a}' \times \mathbf{b}') \cdot (\mathbf{a}' \times \mathbf{c}')}{|\mathbf{a}' \times \mathbf{b}'|\,|\mathbf{a}' \times \mathbf{c}'|} \tag{2.11}$$

Similarly, $\beta$ and $\gamma$ are expressed as follows

$$\beta = \arccos \frac{(\mathbf{b}' \times \mathbf{c}') \cdot (\mathbf{b}' \times \mathbf{a}')}{|\mathbf{b}' \times \mathbf{c}'|\,|\mathbf{b}' \times \mathbf{a}'|} \qquad \gamma = \arccos \frac{(\mathbf{c}' \times \mathbf{a}') \cdot (\mathbf{c}' \times \mathbf{b}')}{|\mathbf{c}' \times \mathbf{a}'|\,|\mathbf{c}' \times \mathbf{b}'|} \tag{2.12}$$

The sign of the signed solid angle $\omega$ is determined by considering the positional relationship from grid point $D$ to triangular facet $ABC$: front side or back side. An arbitrary point $F$ on triangle $ABC$ is introduced as follows

$$\overrightarrow{DF} = \mathbf{f} = -\mathbf{d} + s_1\mathbf{b} + s_2\mathbf{c} \qquad s_1 \geqslant 0 \qquad s_2 \geqslant 0 \qquad s_1 + s_2 \leqslant 1 \tag{2.13}$$

We take the inner product of $\mathbf{f}$ and the unit normal vector $\mathbf{n}$ that is directed to the front of the triangle facet $ABC$

$$\mathbf{f} \cdot \mathbf{n} = (-\mathbf{d}) \cdot \mathbf{n} + s_1\mathbf{b} \cdot \mathbf{n} + s_2\mathbf{c} \cdot \mathbf{n} = (-\mathbf{d}) \cdot \mathbf{n} \tag{2.14}$$

where $\mathbf{f} \cdot \mathbf{n}$ is equal to $(-\mathbf{d}) \cdot \mathbf{n}$ and independent of the position of point $F$ on triangle $ABC$. Thus, the sign of the solid angle $\omega$ is determined as follows

$$\begin{cases} \omega > 0 & \text{if} \quad (-\mathbf{d}) \cdot \mathbf{n} > 0 \quad (D \text{ locates in the back-side of triangle } ABC) \\ \omega < 0 & \text{if} \quad (-\mathbf{d}) \cdot \mathbf{n} < 0 \quad (D \text{ locates in the front-side of triangle } ABC) \end{cases} \tag{2.15}$$

The positional relationship of grid point $D$ and the object surface is estimated by integrating the signed solid angle $\omega$ as follows

$$\sum_{\text{all facet}} \omega = \begin{cases} 0 & \text{if} \quad D \text{ locates outside of the object} \\ 4\pi & \text{if} \quad D \text{ locates inside of the object} \end{cases} \tag{2.16}$$

Considering the dirty STL data, the detection threshold is implemented as follows

$$\begin{cases} \phi > 0: & \text{outside} & \text{if} & \sum\limits_{\text{all facet}} \omega \leqslant 2\pi \\ \phi < 0: & \text{inside} & \text{if} & \sum\limits_{\text{all facet}} \omega > 2\pi \end{cases} \tag{2.17}$$

The estimation of the signed distance function $\phi$ is summarized as follows: 1. Calculate the distance by Eq. (2.7) and Table 1 and 2 determine the sign by Eq. (2.17). In contrast to the estimation of $|\phi|$, the information of all facets is reflected for the estimation of the sign of the signed distance function $\phi$.

## 3. Numerical results of the signed distance function around complex and dirty geometry

The signed distance function around the NASA Common Research Model (NASA-CRM) (Uchiyama *et al.*, 2019), a wind tunnel model of an airplane, is calculated to test the performance of the present method for complex and dirty geometry. The NASA-CRM consists of a body, main wing and tail wing, as shown in Fig. 4a. In this test, robustness to dirty STL data, including gaps, overlaps or inversion of the normal vector of the facets, is evaluated.

Figure 4b shows an example of overlapping facets. Orange, blue and green triangles overlap each other. On the other hand, triangle facets in Fig. 4c are colored by the direction of the normal vector of the facet. Although the main wing consists of one convex surface, some facets are colored red. This suggests an inversion normal vector.

A conventional simple point-in-polyhedron testing method is introduced for comparison. The vector from grid point $D$ to arbitrary point $F$ on the nearest triangle facet is defined as $\mathbf{f}$. Considering the unit normal vector $\mathbf{n}$ is directed outside the object, the sign of the distance function is estimated by the inner product $\mathbf{f} \cdot \mathbf{n}$

$$\begin{cases} \phi > 0: & \text{outside} & \text{if} & \mathbf{f} \cdot \mathbf{n} < 0 \\ \phi < 0: & \text{inside} & \text{if} & \mathbf{f} \cdot \mathbf{n} > 0 \end{cases} \tag{3.1}$$

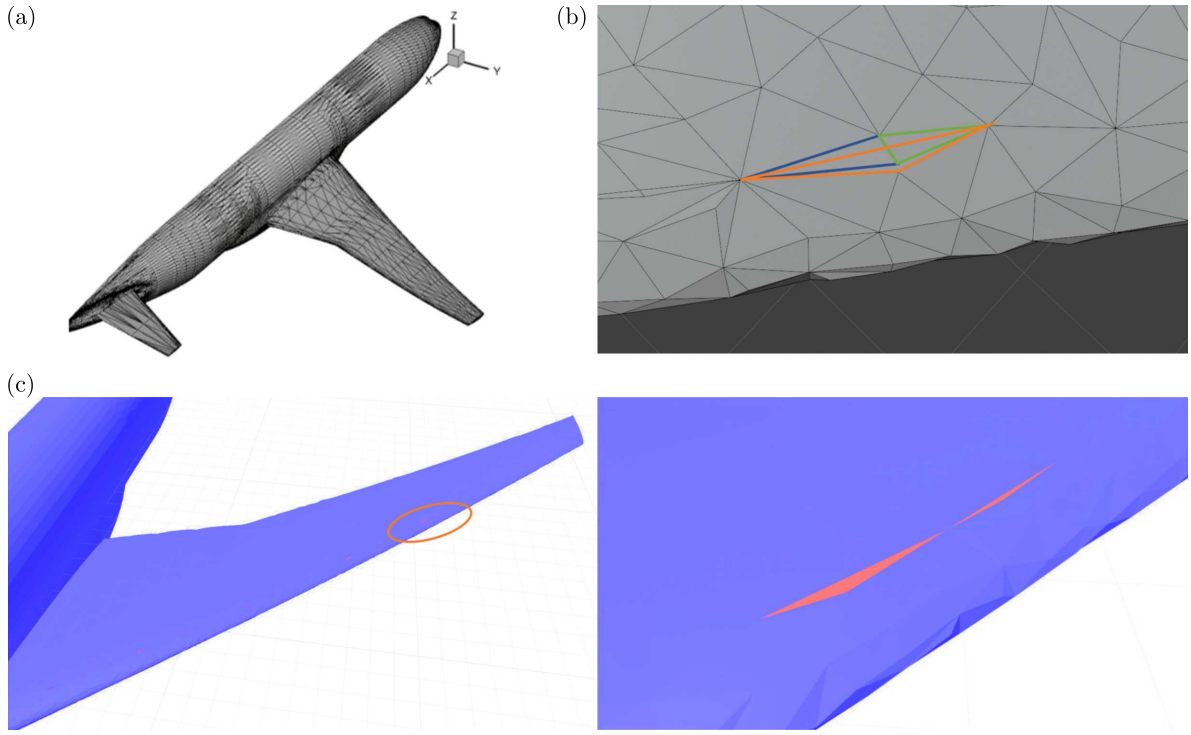


Fig. 4. Goemetry of NASA-CRM and typical errors in the STL data: (a) geometry of NASA-CRM, (b) overlapped triangle facets, (c) triangle facets with the inversion normal vector (red facet), (d) vicinity of the orange circle of the figure (c)

This inner product is independent of the position on the facet, similar to that shown in Eqs. (2.13) and (2.14). Thus, the implementation is as follows

$$\begin{cases} \phi \geqslant 0: & \text{outside} & \text{if} & (-\mathbf{d}) \cdot \mathbf{n} < 0 \\ \phi < 0: & \text{inside} & \text{if} & (-\mathbf{d}) \cdot \mathbf{n} > 0 \end{cases} \tag{3.2}$$

Figure 5 shows the isosurface $\phi = 0$ of the signed distance function for NASA-CRM. In the technique using the summation of solid angles, the sign is correctly determined. However, the trailing edge of the main wing and the tail wing have a saw-like shape. This arises from the grid resolution and has been confirmed to exhibit grid convergence.

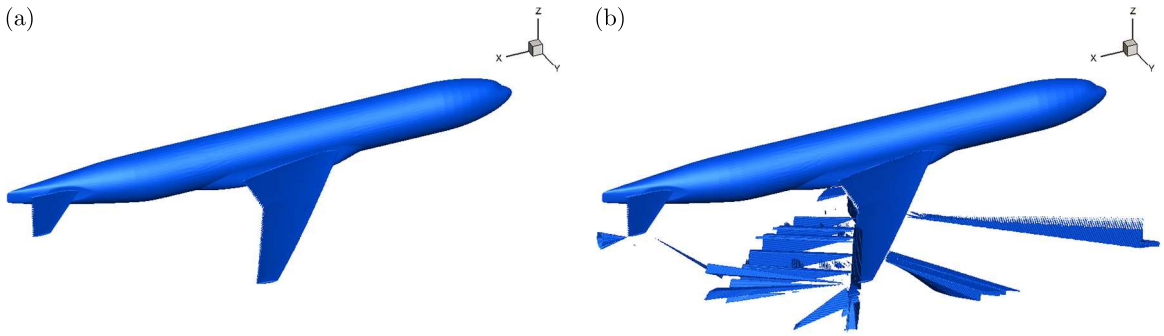


Fig. 5. Distribution of the zero-level isosurface of the signed distance function $\phi$: (a) estimation (2.17) based on spherical trigonometry, (b) conventional estimation (3.2) based on the facet normal

In the result of the technique using the unit normal vector of the nearest facet, there are many grid points where the sign of $\phi$ is misevaluated. When a facet includes inversion of the unit normal vector, the misevaluation of the sign is radially distributed.

Figure 6 shows the errors of volume estimation, and the error is defined as follows

$$\text{Error} = \frac{|V_{\text{Estimated}} - V_{\text{STL}}|}{V_{\text{STL}}} \tag{3.3}$$

Here, $V_{\text{Estimated}}$ is the volume of the NASA-CRM estimated by voxel estimation and cut-cell estimation. The voxel volume estimation is applied to the signed distance function evaluated by the unit normal vector of nearest facet (3.2) and the summation of solid angles (2.17). The cut-cell volume estimation is applied to the signed distance function by the summation of the solid angle. The error of volume estimation of the signed distance function using the summation of the solid angles is converged with a grid refinement. However, the error of the method using the unit normal vector of the nearest facet is independent of the grid resolution.

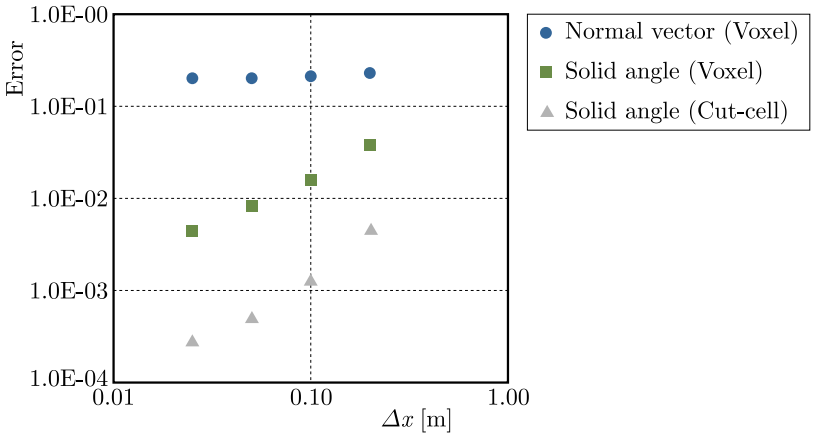


Fig. 6. Errors of volume estimation by two point-in-polyhedron methods: blue circle plots indicate method (3.2) using the unit normal vector of the nearest facet (voxel estimation); green square plots and gray triangle plots indicate method (2.17) using the summation of the solid angles (voxel estimation and cut-cell estimation)

In order to confirm the influence of a facet on the solid angle at an arbitrary grid point, we now calculate the solid angle for a single facet of an equilateral triangle with a side length of 1, and the facet normal vector is $\mathbf{n} = [1, 0, 0]$. It is assumed that the facet is in the $y$-$z$ plane and that the position of the center of gravity coincides with the origin. The maximum solid angle for the facet is obtained in the surface. Figure 7 shows the distribution of the solid angle $\omega$ in the $x$-$y$ cross-section of $z = 0$. The absolute value of the solid angle is $x \to 0$, and the maximum value is $2\pi$. As $|x|$ increases, the value of $|\omega|$ decreases rapidly. Therefore, when the direction of the normal vector of a facet is reversed, its influence on the neighbor grid point is larger, and grid points far from the facet are only minimally influenced.

The value of $\omega$ is negligible at the point shifted from the facet normal direction. Therefore, even when the direction of the normal vector of two or more adjacent facets is reversed, the effects are minimal.

When using the summation of solid angles, the evaluation inversion of the sign occurs if and only if the grid point is extremely close to the facet which exhibits inversion of the normal vector. Even in such a case, the inversion is limited to only one grid point. This indicates the robustness of sign determination (2.17) using the summation of the solid angle when this approach is applied to the data, including the inversion of the facet normal vector.

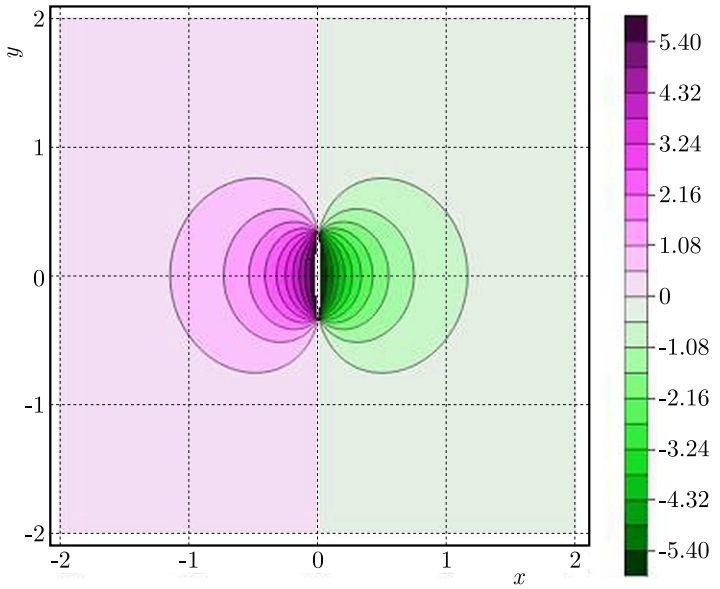


Fig. 7. Distribution of $\omega$ along the facet normal coordinate $x$ and the parallel coordinate $y$

## 4. Numerical example for application to computational fluid dynamics

### 4.1. Calculation conditions

In this study, the calculation condition refers to the experimental study by Uchiyama *et al.* (2019). The object is a half-cut shape of the NASA-CRM composed of the main wing, body, and tail wing. The average aerodynamic chord length $c$ of the main wing is 7.00532 m, and the span length $b$ is 29.38145 m. The angle of attack $\alpha$ has two values: 11.05° and 13.08°. The Mach number of the far uniform flow is set at 0.168. The Reynolds number based on the average aerodynamic chord length is $1.06 \cdot 10^6$. For details of the experimental settings, see Uchiyama *et al.* (2019).

The CFD analysis is performed by the Cartesian cut-cell simulation code developed by Takeda *et al.* (2020a). The Cartesian cut-cell method gives a better volume estimation than the voxel method, as shown in Fig. 6. The calculation area is set to exceed 300 m in both the $x$ and $y$ directions and to exceed 900 m in the $z$ direction. The uniform grid is applied to $2\,\mathrm{m} \leqslant x \leqslant 70\,\mathrm{m}$, $0\,\mathrm{m} \leqslant y \leqslant 32\,\mathrm{m}$, and $2\,\mathrm{m} \leqslant z \leqslant 15\,\mathrm{m}$. The nose of the NASA-CRM places at $x = 2.36\,\mathrm{m}$. The grid width of the uniform grid normalized by chord length $\Delta x/c$ is $7.14 \cdot 10^{-3}$. The implicit large eddy simulation (ILES) is employed in this study, and the wall-stress model by Kawai and Larsson (2012) is used to estimate the wall shear stress.

The inflow boundary condition at the $x$ and $z$ boundaries corresponds to a uniform flow, and the outflow boundary condition is a zero gradient. A symmetric boundary condition is applied at $y = 0$, and the boundary condition at $y = y_{max}$ corresponds to the uniform flow.

### 4.2. Flow around the NASA-CRM

Figure 8 shows the instantaneous distribution of the contour map of the Mach number in an isosurface of the second invariant of the velocity gradient tensor ($Q$-criterion) around the NASA-CRM. A leading-edge separation is observed on the suction side of the main wing along the wide range of the wing span.

Figure 9 shows an oilflow visualization of the experiment (Uchiyama *et al.*, 2019) and the time-averaged surface streamlines in the present study. The blue arrow indicates the origin of the leading edge separation in the spanwise direction. The starting point of the leading edge

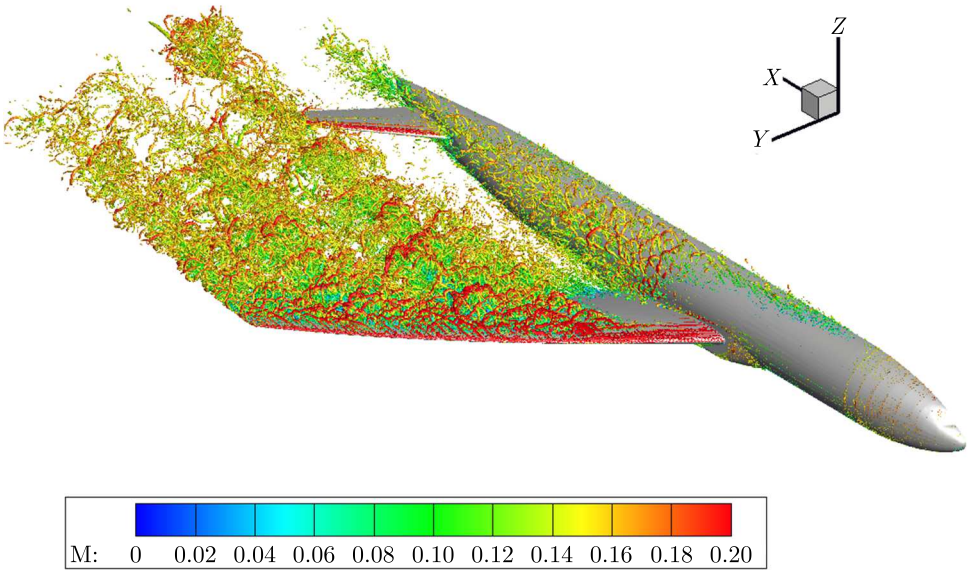


Fig. 8. Instantaneous distribution of the contour map of the Mach number in an isosurface of the second invariant of the velocity gradient tensor at $\alpha = 11.05°$

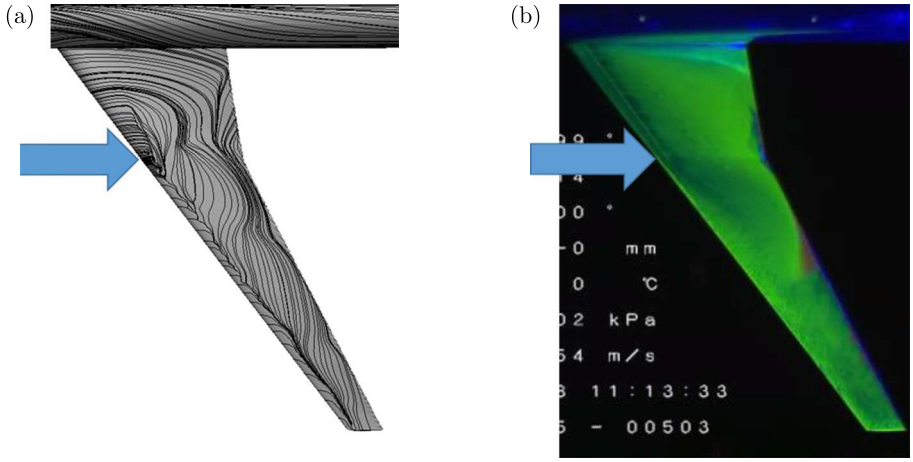


Fig. 9. (a) Surface streamline from the present CFD analysis, (b) oilflow visualization of the experiment (Uchiyama *et al.*, 2019) at $\alpha = 11.05°$

separation agrees with the experimental results. Furthermore, the separation area around the wingtip is also consistent with these results.

### 4.3. Aerodynamic coefficients

The lift coefficient $C_L$, drag coefficient $C_D$, and pitching moment coefficient $C_M$ are calculated using the following expressions

$$C_L = \frac{L}{\frac{1}{2}\rho_\infty U_\infty^2 S} \qquad C_D = \frac{D}{\frac{1}{2}\rho_\infty U_\infty^2 S} \qquad C_M = \frac{M}{\frac{1}{2}\rho_\infty U_\infty^2 Sc} \tag{4.1}$$

Here, $S = 383.6895552\,\mathrm{m}^2$ is the reference area. The constants $\rho_\infty$ and $U_\infty$ indicate the free-stream density and velocity, respectively.

Figures 10a, 10b and 10c show the $C_L$-$\alpha$, $C_D$-$\alpha$ and $C_M$-$\alpha$ diagrams, respectively. The angle-of-attack dependency of the present CFD is consistent with the experimental results, and the error of $C_L$ in the experiment at $\alpha = 11.05°$ is 6.4%. The underestimation of $C_D$ and $C_M$ at $\alpha = 13.08°$ is caused by the underestimation of contribution of the underresolved tail wing.

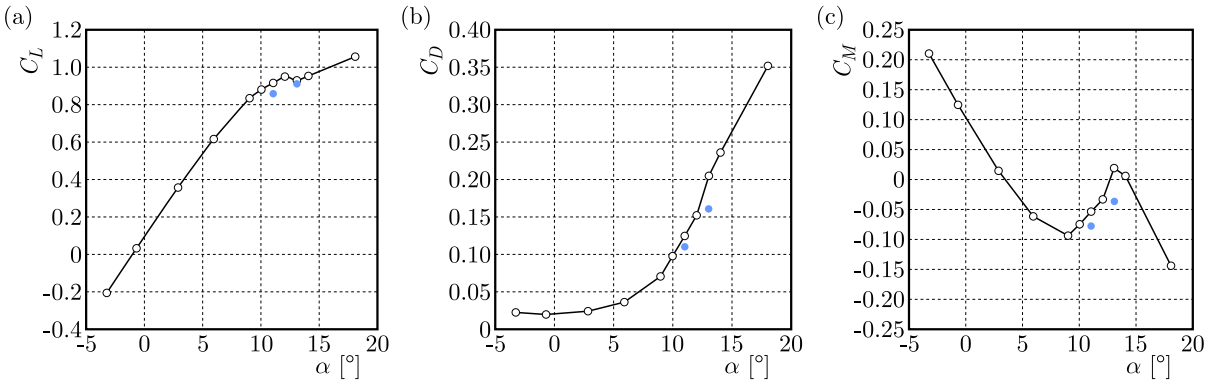


Fig. 10. Aerodynamic coefficients: blue filled circles indicate the present CFD analysis; open circles indicate the experimental results (Uchiyama *et al.*, 2019): (a) $C_L$-$\alpha$ diagram, (b) $C_D$-$\alpha$ diagram, (c) $C_M$-$\alpha$ diagram

## 5. Conclusion

A robust and automatic generation method of the signed distance function for the preprocessing of CFD is proposed. The proposed method uses summation of the solid angle of all facets of the object to ensure robustness for complex and dirty STL data. This generation method of the signed distance function is compared with the conventional method using the facet normal vector of the nearest facet.

In order to test the robustness, the signed distance function is generated from the STL data of the NASA-CRM that contains errors. The summation of the solid angle gives an appropriate signed distance function. However, the conventional method using a facet normal vector of the nearest facet misrecognizes external points of the NASA-CRM as inside points of the object. This misevaluation occurs at the grid point associated with facets with a flipped normal vector.

Computational fluid dynamics simulations of the external flow around the NASA-CRM are conducted to verify the signed distance function generated using the solid angle. This CFD analysis appropriately predicts the starting point of the leading edge separation. The angle-of-attack dependency of the aerodynamic coefficients ($C_L$, $C_D$ and $C_M$) agrees with the experimental results.

*Acknowledgments*

The authors express their profound gratitude to the Cyber Science Center of Tohoku University and Earth Simulator Center of Japan Agency for Marine-Earth Science and Technology (JAMSTEC) for providing computational resources for this research.

## References

1. Baerentzen J.A., Aanaes H., 2005, Signed distance computation using the angle weighted pseudonormal, *IEEE Transactions on Visualization and Computer Graphics*, **11**, 3, 243-253

2. Carvalho P.C.P., Cavalcanti P.R., 1995, *Point in Polyhedron Testing Using Spherical Polygons, Graphics Gems V*, Academic Press, Inc., 42-49

3. Estellers V., Zosso D., Lai R., Osher S., Thiran J.P., Bresson X., 2012, Efficient algorithm for level set method preserving distance function, *IEEE Transactions on Image Processing*, **21**, 12, 4722-4734

4. Hu X.Y., Khoo B.C., Adams N.A., Huang F.L., 2006, A conservative interface method for compressible flows, *Journal of Computational Physics*, **219**, 2, 553-578

5. Iaccarino G., Verzicco R., 2003, Immersed boundary technique for turbulent flow simulations, *Applied Mechanics Reviews*, **56**, 3, 331-347

6. Ishida T., Takahashi S., Nakahashi K., 2008, Efficient and robust Cartesian mesh generation for building-cube method, *Journal of Computational Science and Technology*, **2**, 4, 435-446

7. Kawai S., Larsson J., 2012, Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy, *Physics of Fluids*, **24**, 1, 1-10

8. Lahur P.R., Hashimoto A., Murakami K., 2012, Automatic grid generation for dirty STL data using approximate concave feature, *50th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, January, 1-10

9. Li J., Wang W., 2017, Fast and robust GPU-based point-in-polyhedron determination, *CAD Computer Aided Design*, **87**, 20-28

10. Peskin C.S., 1977, Numerical analysis of blood flow in the heart, *Journal of Computational Physics*, **25**, 3, 220-252

11. Roth S.D., 1982, Ray casting for modeling solids, *Computer Graphics and Image Processing*, **18**, 2, 109-144

12. Sussman M., Smereka P., Osher S., 1994, A level set approach for computing solutions to incompressible two-phase flow, *Journal of Computational Physics*, **114**, 1, 146-159

13. Szilvási-Nagy M., Mátyási G., 2003, Analysis of STL files, *Mathematical and Computer Modelling*, **38**, 7-9, 945-960

14. Takeda Y., Ueno K., Ishikawa T., Takahashi Y., 2020a, Prediction capability of Cartesian cut-cell method with a wall-stress model applied to high Reynolds number flows, *Applied Sciences* (Switzerland), **10**, 15

15. Takeda Y., Ueno K., Matsuyama S., Tanno H., 2020b, Coupled numerical analysis of three--dimensional unsteady flow with pitching motion of reentry capsule – investigation of the third harmonics of the aerodynamic force, *Transactions of the Japan Society for Aeronautical and Space Sciences*, **63**, 6, 249-256

16. Uchiyama T., Kohzai M., Miki H., Hirotani T., Sudani N., Shutoku H., 2019, Experimental investigation of a 160% scaled NASA common research model at low speed conditions, *AIAA Scitech 2019 Forum*, January, 1-21

*Manuscript received January 24, 2023; accepted for print April 18, 2023*