An immersed boundary method for practical simulations of high-Reynolds number flows by $k$-$\varepsilon$ RANS models

Hiroki YAO*, Taisuke NAMBU** and Yasuhiro MIZOBUCHI**

*Ryoyu Systems Co., Ltd., 6-19 Oye-cho, Minato-ku, Nagoya 455-0024, Japan
**Numerical Simulation Research Unit, Aeronautical Technology Directorate, Japan Aerospace Exploration Agency, 7-44-1, Jindaiji-higashi, Chofu, Tokyo 182-8522, Japan
E-mail: mizo@chofu.jaxa.jp

Abstract
A combustion simulation software tool, “HINOCA”, has been developed for automotive engine analysis. HINOCA is based on fully compressible Navier-Stokes equations, which are Reynolds-averaged (RANS) or spatially-filtered (LES), and employs the Cartesian grid and immersed boundary (IB) methods to reduce the mesh generation cost. In the present paper, focusing on flow simulations using $k$-$\varepsilon$ models, a robust and reliable IB method coupled with wall functions is proposed. One major aspect of the method is that different IB cell information is employed for inviscid and viscous flux evaluations at fluid-IB cell interfaces. To improve the evaluation of wall shear stress, the shear stresses on the boundaries of an IB cell are transformed into a body force acting on the adjacent fluid cell. The computational method for $\varepsilon$-equation and the source terms of the $k$-equation near IB cells are modified so that the development of the turbulent boundary layer on a flat plate is well reproduced. The effects of these modifications are validated by the 2D Zero Pressure Gradient Flat Plate problem. To improve the mass conservation property of the IB method, multiple geometric parameters are defined for IB cells; that is, different image point information is immersed on IB cell centers for evaluating the inviscid flux on each cell interface. Evaluation with the Steady State Flow Bench problem shows that the proposed method drastically improves the mass conservation property of simulations and is able with a coarse mesh to reproduce flow structures obtained by LES with a much finer mesh.

Keywords : Immersed boundary method, Reynolds averaged Navier-Stokes, Wall function, Internal combustion engine, Complex geometry

1. Introduction
A simulation software tool for Internal Combustion (IC) engines, “HINOCA”, is currently being developed. IC engines have time-varying geometries such as pistons and valves, and so the cost of mesh generation is an important consideration. To address this issue, HINOCA employs the Cartesian grid and immersed boundary (IB) methods which minimize the computational cost of grid generation, especially for problems that include large deformations. HINOCA is designed for a wide range of uses, from rigorous simulations to study detailed physics in automotive engines (Nambu, et al., 2018) to engineering simulations to be utilized in model-based design (MBD (Mizobuchi, 2018, Mizobuchi, et al., 2019). In the 21st century, combustion simulation with rigorous flow properties and detailed chemical kinetics has become possible for laboratory-size flames (Mizobuchi, et al., 2002, Westbrook, et al., 2005), but such an approach is not still applicable to full scale engine combustion simulation. MBD processes typically use the Reynolds-Averaged Navier-Stokes (RANS) approach since its computational cost is much lower than those of the Large Eddy Simulation (LES) and Direct Numerical Simulation (DNS) approaches. In particular, in applications that require many trial simulations, RANS is used with coarse meshes that cannot resolve the internal structure of boundary layers. HINOCA employs wall function models for such cases where a boundary layer-resolving mesh cannot be used.
The “immersed boundary method” was originally developed by Peskin (1972) to simulate laminar flows around human heart valves on a rectangular mesh by introducing a forcing term. For high Reynolds number flows, the ghost-cell approach has been proposed, as is detailed in (Mittal and Iccarino, 2005). Although these conventional IB methods work well with sufficiently fine meshes, they were not designed for simulations using wall functions on coarse meshes. In IC engine development, RANS simulations with wall functions are run on coarse meshes. Therefore, the present paper proposes an IB method coupled with RANS models and wall functions. For internal flow simulations, $k$-$\varepsilon$ models are often employed as RANS models, and therefore $k$-$\varepsilon$ models are treated in the present paper.

During its early phase of development, HINOCA was based on ghost-cell IB methods coupled with RANS models and wall functions. However, the results were less satisfactory than those of other software tools using boundary fitted meshes. To improve its performance, three technical modifications were applied: imposition of boundary conditions on immersed boundaries; near-wall treatments of $k$-$\varepsilon$ models; and treatment of convex boundaries. The first and second points aimed to improve the accuracy and stability of calculations of turbulent boundary layers, and their effectiveness was evaluated using the 2D Zero Pressure Gradient Flat Plate problem (NASA, Turbulent Modeling Resource). The last term is to improve mass conservation around convex boundaries, and its effectiveness was demonstrated with a simple backward-facing step flow. To show its applicability to a practical internal flow simulation, the proposed method was evaluated with a well-known complex geometry, the Steady State Flow Bench problem (Thobois, et al., 2004).

2. Immersed boundary methods

Figure 1 (a) shows a summary of conventional IB methods. Computational cells are classified as fluid, solid or IB cell. On fluid cells, the system of compressible fluid equations is solved. IB cells are placed near boundaries and serve as “ghost cells” for the computation of adjacent fluid cells, allowing the numerical flux at fluid-IB cell interfaces to be computed in the same manner as at fluid-fluid cell interfaces. The fluid variables at an IB cell are expressed using information at an interior point and conditions imposed on the boundary. The interior point is called an “image point” and is set at a distance along the normal direction from the boundary.

The present paper uses the ghost-cell approach. The equations are discretized on a Cartesian mesh with a uniform cell size $\Delta x$ throughout. The inviscid terms of the equations are evaluated by SLAU2 (Kitamura and Shima, 2010). The inviscid flux at fluid-IB cell interfaces is evaluated with 1st-order accuracy. For regions more than two cell diameters from boundaries, 3rd-order spatial accuracy is achieved using a MUSCL scheme (van Leer, 1979) with the van Albada limiter (van Albada, 1982).

2.1 Location of IB cells

When using an IB method with a ghost-cell approach, IB cells (that is, ghost cells) are located around body surface boundaries, and there are several ways in which such cells can be defined. One of the simplest definitions is that cells that are intersected by boundaries are defined as IB cells, in which case the center of an IB cell can be in either the fluid zone or the solid zone. Another definition is that an IB cell is a cell whose center is located outside of body surface

![Fig. 1 Outline of IB methods: Each cell is classified as a fluid (white), solid (gray) or IB cell (red). $B$, $P$ and $W$ are respectively the center of IB cell, image point and wall point where boundary conditions are imposed.](image-url)
boundaries, in which case the center of an IB cell is always in the solid zone (see Fig. 1 (b)). The latter definition is employed for our IB methods because it reproduces flow fields better than the former when valve lift is low.

2.2 Initial approach

In the first approach used in the development of HINOCA, wall functions were utilized to embed the velocities at IB cells, and wall shear stresses were derived from wall functions and fluid values at the image points. The same embedded values were used for both viscous and inviscid flux calculations at fluid-IB interfaces. This method is employed in conventional IB methods. The velocity imposition and near-wall treatments of $k$-$\varepsilon$ models in the initial approach are described below.

2.2.1 Velocity imposition

With this method, the embedded velocity at an IB cell is determined from the following relationship between the wall shear stress $\tau_w$ and the boundary-normal gradient of tangential velocity:

$$\tau_w = (\mu + \mu_t) \frac{\partial u}{\partial y},$$

where $y$ is “height” (distance in the wall normal direction) from the boundary. $\mu$ and $\mu_t$ are the molecular viscosity and the turbulent viscosity respectively. Note that $\tau_w$ is evaluated by a wall function using variables at the image point.

2.2.2 Near-wall treatment of $k$-$\varepsilon$ models

The production term of the $k$-equation is expressed as

$$P_k = \mu_t S^2,$$

$$S \equiv 2S_{ij}S_{ij},$$

where $S_{ij}$ is the rate-of-strain tensor

$$S_{ij} = \frac{1}{2} \left( \frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right).$$

For bulk regions far from the boundary, the gradients in Eq. (4) are calculated at cell centers by a central difference scheme. On the other hand, to estimate the production term on fluid cells that are adjacent to IB cells, the gradients in Eq. (4) are calculated by the one-sided difference between the fluid and IB cells. Except for the evaluation of the gradient, the $k$-equation is evaluated in the same manner in the near-wall and bulk regions.

For the wall boundary conditions of $k$ and $\varepsilon$-equations, it is assumed that the image point is located in the log-law region. While the boundary condition of $k$ is given as a Dirichlet type, the boundary condition of $\varepsilon$ is given as a Neumann type:

$$k = \frac{\tau_w/\rho}{C_{\mu}^{1/2}},$$

$$\frac{\partial \varepsilon}{\partial y} = \frac{\partial}{\partial y} \left( \frac{k^{3/2}}{\rho y C_{\mu}^{3/4}} \right).$$
where \( \kappa \) is the Karman constant and is set to 0.41.

### 2.2.3 Results for 2DZP problem

The initial method was evaluated using the 2D Zero Pressure Gradient Flat Plate (2DZP) problem provided by the National Aeronautics and Space Administration (NASA, Turbulence Modeling Resource). Figure 2 shows the problem description of 2DZP. This case is run at \( M = 0.2 \) at a Reynolds number of \( Re = 5 \times 10^6 \). Two geometric conditions were employed: one in which the boundaries of 2DZP and cell interfaces coincided; and one in which they did not coincide and the computation domain was rotated left by 15 degrees. Since not-coincident boundary and cell interfaces and tilted boundaries exist in practical simulations, it is required that predictions near such boundaries are solved accurately. The cell size \( \Delta x \) was set to \( 0.5 \times 10^{-3} \) in non-dimensional units, corresponding to \( y^+ \sim 90 \) at \( X=0.97 \). The distance between image points and the boundary was set to \( 0.5 \Delta x \) for this calculation. Table 1 shows a detailed summary of the analytical methods used. Results computed by SST-V (Menter, 1992) of CFL3D (NASA’s software) are given for reference. CFL3D was used with a boundary-fitting mesh, and the height of the first layer mesh corresponds to \( y^+\sim 1 \).

For the unrotated domain, a turbulent layer develops and the predictions of fluid variables qualitatively agree with the reference results (See Fig. 3). Note that Fig. 3 and the following Figs. 4 and 5 include “modified” results, which are explained later. However, \( k \) does not develop immediately after the edge since the production \( P_k \) is almost zero (Fig. 4 (a)). Although \( P_k \) is almost zero after the edge, \( k \) gradually increases due to the boundary condition Eq. (5), and these behaviors should be modified. For the 15-degree rotated domain, there are several issues. First, the velocity near the plate is lower than the reference (See Fig. 5 (c)). Second, the wall shear stress fluctuates along the plate, and is much lower than the reference (Fig. 5 (a)) due to the lower velocities formed near the wall. Third, the distribution of \( k \) near the plate does not obey its log-law profile \( d k / d y = 0 \) (Fig. 5 (e)). Finally, distribution of \( \varepsilon \) near the plate does not obey its log-law profile \( \varepsilon \propto 1/y \) (Fig. 5 (f)).

![Fig. 2 Computational configuration of 2D Zero Pressure Gradient Flat Plate problem (NASA, Turbulence Modeling Resource).](image)

| Table 1 | Summary of numerical methods for 2DZP. |
|---------|----------------------------------------|
| Governing equation | Compressible RANS |
| Inviscid flux | SLAU2 (Kitamura and Shima, 2010) |
| | 3rd-order MUSCL (van Leer, 1979) with van Albada limiter (van Albada, 1982) |
| Viscid flux | 2nd-order central difference |
| Time integration | LUSGS (Jameson and Yoon, 1987) with Preconditioner (Weiss and Smith, 1995) |
| Turbulence model | Standard \( k-\varepsilon \) (Launder and Spalding, 1972) |
| Wall function | Spalding’s law type (Knopp, 2006) |

[DOI: 10.1299/jfst.2021jfst0007] © 2021 The Japan Society of Mechanical Engineers
Fig. 3 2DZP results for unrotated (0-degree tilted) domain.
2.3 Modified method

In the following discussions, we clarify the causes of the problems found with the initial approach and propose modifications. The modified method was then evaluated by 2DZP as with the initial method.

2.3.1 Velocity imposition

As shown in Fig. 5 (c), the velocity near the plate was lower than the reference. This is because very small velocities were embedded through Eq. (1) at the IB cells around the edge of the plate, and this information was then propagated in the streamwise direction by momentum advection. To rectify this and obtain the log-law velocity profile accurately, the viscous and inviscid terms around IB cells were calculated using different IB information. Figure 6 shows treatment of the viscous term at a fluid cell (hatched cell) adjacent to any IB cell. The viscous flux at fluid-fluid interfaces $F_3$ and $F_4$ are calculated in the same way as other fluid-fluid interfaces. On the other hand, it is assumed that the viscous flux at fluid-IB interfaces $F_1$ and $F_2$ are identical to the wall shear stress flux $\tau_w$. The wall shear stress flux at each fluid-IB interface is distributed from the neighboring IB cell (point $B$) according to the projected area $\Delta S_F \cdot n_F$ when the cell-interface area vector $\Delta S_F$ is projected toward the wall normal $n_F$. The distributed shear stress fluxes at each fluid-IB interface are then summed and converted into a body force working on the fluid cell (See Fig. 6). Therefore, the body force $F_b$ acting on the fluid cell is calculated as follows:

$$F_b = \frac{1}{\Delta V} \sum_{F \in \text{fluid-IB}} (\tau_w)_F \times (\Delta S_F \cdot n_F), \quad (7)$$

where $F$ denotes an interface and $\Delta V$ is the cell volume.

For the inviscid term at a fluid cell adjacent to any IB cell, the inviscid flux at each fluid-fluid interface is calculated in the same way as other fluid-fluid interfaces. To calculate the inviscid flux at each fluid-IB interface, the velocity at each IB cell is evaluated with the symmetric boundary condition

$$\frac{\partial u}{\partial y} = 0. \quad (8)$$

These modifications eliminate extreme embedded velocities in IB cells due to Eq. (1), and it is expected that calculations will be more stable than those of the initial method.
Fig. 5 2DZP results for 15-degree rotated domain.
As shown in Fig. 5 (a), the shear stress along the plate computed by the initial method is not accurate. This is because the velocity at an image point is 0th-order extrapolated from the nearest cell center, so the height of the point above the plate and the extrapolated velocity are inconsistent. To remedy this problem, the tangential velocity $u$ at the image point is reconstructed. The reconstructed function is assumed to be a linear function:

$$u = c_1 n + c_2 t + c_3,$$  \hspace{1cm} (9)

where $n$ and $t$ are the wall-normal and -tangential coordinates, respectively, with the origin at the wall point $W$ (See Fig. 7). Coefficients $c_1$, $c_2$ and $c_3$ are determined by a least squares method among the fluid cells surrounding the point $P$. This modification improves the evaluation of the shear stress in Eq. (7).

Fig. 6 Treatment of the viscous term at fluid cell adjacent to any IB cell: Velocity profile is expressed by the blue vector and curved line. The hatched cell is a fluid cell. $F$, $B$ and $P$ are the cell interface, IB cell center, and image point respectively. Wall shear stress flux $\tau_w$ is defined at $B$ using a wall function and values at $P$. $\Delta S_F$ and $n_F$ are cell-interface area vector and wall normal respectively. Body force $F_B$ is sum of forces on fluid-IB interfaces.

Fig. 7 Local coordinates $(n, t)$ and fluid cells (hatched) to be used for the tangential velocity reconstruction by a least squares method.
2.3.2 Near-wall treatment of k-ε models

As shown in Fig. 5 (e), the near-wall behavior of turbulent kinetic energy \( k \) does not qualitatively agree with that obtained by CFL3D. Additionally, as pointed out above, \( P_k \) cannot be computed accurately because turbulent viscosity \( \mu_t \) is yet to develop. To address this issue, that is to predict near-wall \( P_k \) correctly, relations between \( P_k \) and the wall shear stress \( \tau_w \) which hold in the log-law region are employed. However, \( \tau_w \) information is not available at fluid cells because the information is stored only at IB cells. \( \tau_w \) information defined at IB cells is therefore propagated into the fluid region by following equation:

\[
\frac{\partial \tau_w}{\partial t_p} + \mathbf{n} \cdot \nabla \tau_w = 0,
\]

where \( t_p \) is pseudo time and \( \mathbf{n} \) is the normal vector to the boundary. This equation is calculated for near-wall region at each interval of the fluid integration, and the calculation is done simply by using the Euler explicit scheme in pseudo time and the 1st-order upwind scheme in space with several iterations. After the \( \tau_w \) information defined at IB cells is propagated, it becomes available at near-wall fluid cells. Then, the following well-known expression for the production term of \( k \) in the log-law region (Launder and Spalding, 1974) can be employed:

\[
P_k = \frac{\tau_w^2}{\kappa C_{\mu}^{1/4} \rho k^{1/2} y}.
\]

However, the height \( y \) at a fluid cell center can be almost zero because of the locations of IB cells defined as described in subsection 2.1. For a numerical reason, Eq. (11) cannot be applied to such cells because the term can take extremely high values. Additionally, points where \( y \) is low can be in the viscous sublayer. To deal with these issues, the two-layer approach proposed by Chieng and Launder (1980) is introduced. Since the proposed formulations are for the first layer of boundary-fitting meshes, the formulations were modified to be applied to fluid cells adjacent to wall boundaries as well as to distant fluid cells as follows:

\[
P_k = \begin{cases} 
\frac{1}{\Delta x} \int_{y_1}^{y_2} \frac{\tau_w^2}{\kappa C_{\mu}^{1/4} \rho k^{1/2} y} dy & \text{if } y_1 > y_\nu \\
\frac{1}{\Delta x} \int_{y_\nu}^{y_2} \frac{\tau_w^2}{\kappa C_{\mu}^{1/4} \rho k^{1/2} y} dy & \text{otherwise}
\end{cases}
\]

\[
\bar{\epsilon} = \begin{cases} 
\frac{1}{\Delta x} \int_{y_1}^{y_2} \frac{k^{3/2}}{\kappa C_{\mu}^{-3/4} y} dy & \text{if } y_1 > y_\nu \\
\frac{1}{\Delta x} \left( \frac{2 \nu k}{y_\nu^2} + \int_{y_\nu}^{y_2} \frac{k^{3/2}}{\kappa C_{\mu}^{-3/4} y} dy \right) & \text{otherwise}
\end{cases}
\]

where the subscripts 0, 1 and 2 denote the fluid cell center, lower bound interval, and the upper bound of the integral interval respectively. That is, \( y_1 \) and \( y_2 \) are defined as \( y_1 = y_\nu - \Delta x / 2 \) and \( y_2 = y_\nu + \Delta x / 2 \). \( y_\nu \) is thickness of the viscous sublayer which is determined by \( y_\nu^* = y_\nu k^{1/2} / \nu \), where \( y_\nu^* \) is the non-dimensional thickness of the sublayer and is set to 11.225 in the present paper. Finally, the blended production term \( P_{k,\text{blend}} \) is derived by blending the averaged production term Eq. (12) for the near wall region and the production term of bulk region \( P_{k,\text{bulk}} \):

\[
P_{k,\text{blend}} = \varphi(y) P_{k,\text{bulk}} + (1 - \varphi(y)) P_k,
\]

where \( \varphi \) is a blending function. In this study, following formula was used:

\[
\varphi(y) = \begin{cases} 
0 & \text{if } y < y_1 \\
\frac{(y - y_1)}{(y_2 - y_1)} & \text{if } y_1 \leq y \leq y_2 \\
1 & \text{if } y > y_2
\end{cases}
\]
\[
\varphi = \frac{1}{2} \left\{ \exp[10 \times (y/\Delta x - 1.5)] - 1 \right\} + 1.
\]

This function is designed so that \( \varphi \approx 0 \) for \( y < \Delta x \), \( \varphi = 0.5 \) at \( y = 1.5\Delta x \) and \( \varphi \approx 1 \) for \( y > 2\Delta x \). For the wall boundary condition of \( k \), \( dk/dy=0 \) is applied.

As shown in Fig. 5 (f), the near-wall \( \varepsilon \) does not follow the expected profile. Originally, near-wall \( \varepsilon \) decreases significantly near the plate. In this case, \( \varepsilon \) drops sharply within a few cells along the wall direction. Obviously, transport of \( \varepsilon \) with such an extremely high gradient for the near-wall region cannot be solved accurately due to discretization error. In particular, that of the advection term is considerable because the stream is not along a mesh axis. To address this issue, near-wall \( \varepsilon \) was enforced by Eq. (13) where \( y_0 < 2\Delta x \), since numerically solving its transport equation is difficult when meshes are not adequately fine. The same measure was also applied for the destruction term of the \( k \)-equation.

### 2.3.3 Results and discussion for 2DZP problem

The modified method was evaluated again using the 2DZP problem. For the unrotated domain, the modified method reproduces fluid variables that are comparable to the reference values similarly to the initial method (Fig. 3). Around the plate edge, the modified near-wall treatment of \( k-\varepsilon \) models improved the behaviors of turbulent kinetic energy \( k \) and its production term \( P_k \) such that they develop immediately after the edge (Fig. 4 (b)).

For the 15-degree rotated domain, significant several improvements were obtained (Fig. 5). First, the velocity near the plate becomes closer to the reference (Fig. 5 (c)). This is because the inviscid flux at fluid-IB interfaces is evaluated applying the symmetric boundary condition Eq. (8) to the plate, so no extremely small velocities at IB cells harm the distribution above the plate. While the symmetric boundary condition is applied for the inviscid flux at fluid-IB interfaces using the ghost-cell approach, the viscous flux at fluid-IB interfaces is considered by Eq. (7) so that the wall shear stress works there. These modifications contribute to the accurate formation of the velocity profile above the plate, as indicated in Fig. 5 (c). Second, the wall shear stress along the plate shows a smoother curve than the initial method (Fig. 5 (a)) since the velocities at the image points are reconstructed. Third, the behaviors of \( k \) and \( P_k \) near the edge of the plate are improved, that is \( k \) and \( P_k \) develop from the vicinity of the edge (Fig. 5 (e)). Fourth, the \( \varepsilon \) distribution above the plate is improved (Fig. 5 (f)) in that \( \varepsilon \) in the log-law region obeys the relation \( \varepsilon \propto 1/y \) with the modified method, while it did not with the initial method. Finally, turbulent viscosity \( \mu_\text{t} = \rho C_s k^2/\varepsilon \) is improved (Fig. 5 (b)) since \( k \) and \( \varepsilon \) are improved above the plate.

Although the above improvements were obtained for the 15-degree rotated domain, there are still several differences compared to the reference: the velocity just above the plate is higher (Fig. 5 (d)); the wall shear stress along the plate is greater (Fig. 5 (a)); and \( k \) does not strictly satisfy \( dk/dy = 0 \) (Fig. 5 (e)). The first discrepancy causes other two: the higher the velocity on the plate, the greater the shear stress becomes, resulting in the production of higher \( P_k \) and \( k \) tending to increase. The symmetric boundary condition Eq. (8) for the inviscid flux at fluid-IB interfaces can be considered as one reason why the velocity is higher just above the plate. The velocity seems to just obey the boundary condition Eq. (8), and should be modified in a future work.

### 2.4 Treatment of convex boundaries

Zero-mass flux perpendicular to the body surface boundaries cannot be explicitly imposed in IB methods. Conservation might be poor around convex boundaries if the boundary is not treated carefully. In this section, mass conservation is discussed with a backward-facing step flow as a typical example of a convex boundary.

#### 2.4.1 Conservation errors around a convex boundary and principle strategy for improvement

In present IB methods, each IB cell has an image point and a normal vector. While this works well for a flat boundary, for a boundary which is convex towards fluid zone, local conservation errors around the boundary increase for coarse meshes. This is because the velocity around the boundary is not so different from the bulk velocity when the
mesh is not fine enough to resolve the viscous sublayer. Let us consider mass conservation with a backward-facing step flow, shown in Fig. 8 (a). A uniform velocity $u_\infty$ is assumed before the step to ensure that the velocity around the boundary and that of bulk are the same. The image point $P$ is set above an IB cell with the boundary normal vector pointing upward. The embedded velocity vector at the IB cell $u_B$ is $(u_\infty, 0)$ because of the boundary condition in Eq. (8). Concerning mass conservation error around the IB cell, the mass flux $\dot{m}_{F_2}$ is zero but the flux $\dot{m}_{F_1}$ is $\rho u_\infty$. This means that the mass increases immediately when the flow passes through the backward-facing step. If the image point $P$ is placed at right-hand side of the IB cell and the normal vector points rightward, the mass flux $\dot{m}_{F_2}$ is zero since the normal velocity of the interface $F_2$ is zero. While the zero-mass flux is achieved at $F_2$, that of the $F_1$ interface is not imposed. This indicates that each interface should have its own separate image point and normal vector. In Fig. 8 (b), the IB cell near the convex boundary has an image point $P$, a boundary normal vector, and a ghost cell $B$ for each interface $F$. Then $\dot{m}_{F_1}$ and $\dot{m}_{F_2}$ are zero in this case.

![Fig. 8 Mass conservation error around a convex boundary.](image)

(a) Without treatment of convex boundaries  
(b) With treatment of convex boundaries

2.4.2 Definitions of multiple image points and wall-normal vectors

For convex boundaries, multiple image points are defined so that the image point for each fluid-IB interface is placed in a fluid cell adjacent to the interface. Fluid variables at such an image point are 0th-order extrapolated from fluid cell center (See Fig. 9).

Multiple normal vectors at IB cells are calculated from a signed distance function $\phi$ (or level-set field) where $\phi = 0$ at the boundary, $\phi > 0$ for the fluid zone and $\phi < 0$ for the solid zone. According to the level set method context, the boundary normal vector is evaluated as $n = \nabla \phi / \sqrt{\phi_x^2 + \phi_y^2}$ where $\nabla \phi = (\phi_x, \phi_y)$. For multiple normal vectors, each vector is calculated at the fluid cell center adjacent to face of the IB cell (See Fig. 9).

This series of procedures is applied to IB cells located near a convex boundary. It is therefore required to distinguish a convex boundary from a planar one. In order to detect and extract convex boundaries, curvature $\kappa$ is used: $\kappa = \nabla \cdot n = (\phi_x \phi_y^2 - 2 \phi_x \phi_y \phi_y + \phi_x^2 \phi_y^2)/(\phi_x^2 + \phi_y^2)^{3/2}$ where $\phi_{kk} (k=x, y)$ is the second-order derivative of $\phi$. Since the signed distance function $\phi$ is greater than zero in the fluid zone, the curvature is greater than zero when the boundary is convex toward the fluid zone. The derivatives are computed numerically.
2.4.3 Results and discussion for backward-facing step flow

The above treatment of convex boundaries was evaluated with a simple backward-facing step flow. The wall boundary configuration around the step was as shown in Fig. 8. A coarse mesh was used for the test, with 10 cells before the step and 20 cells after the step in the wall direction. The center of the first cell from the wall does not lie in the viscous sublayer. Figure 10 shows the mass flow rate error along the longer direction. The step is placed at the origin. The figure indicates that the mass increase without the treatment of convex boundaries is 9% at the step. Without the treatment, mass conservation error $E_{\text{mass}}$ can be predicted as $E_{\text{mass}} = 1 / n_{\text{cell}}$, where $n_{\text{cell}}$ is the number of cells aligned before the step (See Fig. 8 (a)). In this case, the error is predicted as $E_{\text{mass}} = 1/10$, and this agrees with the mass increase of the result without the treatment in Fig. 10. The mass increase is caused by an IB cell placed at the corner of the step which has only one image point and a wall-normal vector, as in Fig. 8 (a). On the other hand, the mass does not increase at the step when the treatment of convex boundaries is applied due to the multiple image points and normal vectors at the corner IB cell.

![Fig. 9 Definitions of multiple image points and wall-normal vectors. A fluid cell adjacent to interface $F$ is employed for an image point $P$ to calculate flux at $F$. Wall-normal vectors $n$ are calculated using signed distance function $\phi$.](image)

![Fig. 10 Mass conservation error for a backward-facing step.](image)
3. A practical internal flow simulation

In this section, the proposed IB method is evaluated with a 3D problem: the Steady State Flow Bench of Thobois, et al. (2004).

3.1 Problem description of the Steady State Flow Bench

The Steady State Flow Bench configuration is a sudden expansion pipe with a ratio of 3.5 and a valve. Figure 11 shows a two-dimensional view of the configuration including the symmetric axis with geometrical parameters. The operating fluid is nitrogen gas, which flows from the left side of the picture at a mass flow rate of 0.055 kg/s.

Since the configuration is axis-symmetric, a 1/4-sectored computational domain is used in the present paper. As boundary conditions, the prescribed mass flow rate is given at the inlet boundary and the static pressure is fixed at the outlet boundary. The sectored face is treated as symmetry boundary and the wall function is applied for other walls with no heat flux. The analysis method is the same as shown in Table 1 except for the turbulence model. In this analysis, RNG $k$-$\varepsilon$ (Yakhot and Orszag, 1986) is employed as a turbulence model, which is often used for IC engine calculations. The analysis is done for $\Delta x = 0.25$ mm.

3.2 Results and discussion for Steady State Flow Bench

Figure 12 shows the mass flow rate along the axis of the computational domain (the ordinate is in non-dimensional units). Note that the nominal mass flow rate is 0.01375 kg/s since a 1/4-sectored computational domain is used. The treatment of convex boundaries drastically improves the accuracy of the computed mass flow rate: the increase in mass flow rate between the inlet and the outlet boundary is 1.64% without the treatment, while it is 0.06% with the treatment applied.

There are two locations where the mass flow rate increases abruptly in the case without the convex boundary treatment. The first increase in mass flow rate ($x/R=-0.2$) is detected where a backward-facing step shape is at the left edge of the computational domain (Fig. 11). This improvement is achieved in the same way as described in subsection 2.4.3. The second location is at the valve tip ($x/R=0.16$). The convex boundary treatment gives a lower increase in mass flow rate there, which indicates that it also works well at sharp convex boundaries.

To evaluate the effect of the treatment of convex boundaries on the overall flow field, our results were compared to an LES analysis of the Steady State Flow Bench problem with a very fine mesh system (Nambu, et al., 2018) using block-based adaptive mesh refinement (Matsuo, et al., 2012). Figure 13 shows the Mach number field around the valve. In the reference analysis, the flow separates at the first convex shape on the outer wall (Fig. 13 (a)), and the cell size $\Delta x$ is set to 0.125 mm there. In our analyses with $\Delta x$ set to 0.25 mm uniformly, the flow separation does not occur without the convex boundary treatment (Fig. 13 (b)), but does when the treatment is applied (Fig. 13 (c)). By introducing the treatment, our proposed method is able to reproduce flow features in the reference analysis despite a coarser mesh.
Fig. 12 Mass flow rate for Steady State Flow Bench computation.

(a) LES with finer mesh (time-averaged) (Nambu, et al., 2018)

(b) Present result without treatment of convex boundaries

(c) Present result with treatment of convex boundaries

Fig. 13 Mach number field around the valve for Steady State Flow Bench problem.
4. Conclusion

In the present paper, an IB method coupled with RANS models and wall functions is proposed. An initially used method based on the conventional IB methods was examined with a 2DZP problem, revealing the following issues: Extremely small velocities can be embedded in IB cells and make simulations unstable; Turbulent variables above the flat plate do not obey their profile in log-law region; Turbulent kinetic energy $k$ and its production $P_k$ do not develop immediately after the plate edge. To improve the stability of calculations, the velocity imposition method on immersed boundaries was modified such that different IB cell information is employed for inviscid and viscous flux evaluations at fluid-IB cell interfaces. Also, near-wall treatments of $k$-$\varepsilon$ models were modified to improve the reproducibility of turbulent variables: the blended production term $P_{k,\text{blend}}$ is employed; and transport of $\varepsilon$ is not solved but imposed algebraically instead. The modified method showed significant improvements for the 2DZP problem, but the symmetric boundary condition for the inviscid term appeared to cause a higher velocity just above the plate compared to the reference, which consequently caused the wall shear stress along the plate to be overestimated. The condition requires modification in future work.

For IB cells around convex boundaries, it was found that a single image point and wall-normal vector might make mass conservation error so large as to degrade the global solution when meshes are coarse. To improve mass conservation, multiple image points and wall-normal vectors are defined for IB cells adjacent to convex boundaries, and the inviscid flux calculation at each fluid-IB interface uses its own image point and wall-normal vector. To extract such IB cells and to define the multiple normal vectors, the signed distance function around the boundary is utilized.

The proposed IB method was applied to a practical internal flow problem, the Steady State Flow Bench problem. It was found that multiple image points and wall-normal vectors work well for a sharp convex boundary such as a valve tip. The treatment of convex boundaries is able to reproduce with a coarse mesh flow structures obtained by LES with a very fine mesh system.

The proposed methods extend applicability of IB methods to IC engine simulations that employ coarse meshes and wall functions.

Acknowledgements

This work was supported by the Cross-ministerial Strategic Innovation Promotion Program (SIP). Computations in this paper were performed on the JAXA Supercomputer System generation 2 (JSS2).

References

Chieng, C. C. and Launder, B. E., On the calculation of turbulent heat transport downstream from abrupt pipe expansion, Numerical Heat Transfer, Vol. 3 (1980), pp.189-207.
Jameson, A. and Yoon, S., Lower-upper implicit schemes with multiple grids for the Euler equations, American Institute of Aeronautics and Astronautics Journal, Vol.25, No.7 (1987), pp.929-935.
Kitamura, K. and Shima, E., Improvements of simple low-dissipation AUSM against shock instabilities in consideration of interfacial speed of sound, Proceedings of ECCOMAS CFD 2010 (2010), Paper No. 1283.
Knopp, T., Alrutz, T. and Schwamborn, D., A grid and flow adaptive wall-function method for RANS turbulence modelling, Journal of Computational Physics, Vol.220 (2006), pp.19-40.
Launder, B. E., and Spalding, D. B., Lectures in mathematical models of turbulence, Academic Press, London, England (1972).
Launder, B. E., and Spalding, D. B., The numerical computation of turbulent flows, Computer Methods in Applied Mechanics and Engineering, Vol.3 (1974), pp.269-289.
Matsuo, Y., Kuwabara, T. and Nakamori, I., A parallel structured adaptive mesh refinement approach for complex turbulent shear flows, Journal of Fluid Science and Technology Vol.7 (2012), pp.345-357.
Menter, F. R., Improved two-equation $k$-omega turbulence models for aerodynamic flows, NASA Technical Memorandum (1992), Paper No.103975.
Mittal, R. and Iaccarino, G., Immersed boundary methods, Annual Review of Fluid Mechanics, Vol.37 (2005), pp.239-261.
Mizobuchi, Y., Tachibana, S., Shinjo, J., Ogawa, S. and Takeno, T., A numerical analysis of the structure of a turbulent hydrogen jet lifted flame, Proceedings of the Combustion Institute Vol. 29 (2002), pp. 2009-2015.

Mizobuchi, Y., Automotive engine combustion simulation software “HINOCA,” Journal of Society of Automotive Engineers of Japan Vol.72 (2018), pp105-109 (in Japanese).

Mizobuchi, Y., Nambu, T. and Yao, H., Platform of combustion simulation software HINOCA, 16th International Conference on Flow Dynamics (2019), Paper No.5473-1.

Nambu, T., Mizobuchi, Y., Kuwabara, T. and Kiriwara, R., Applying high-order compact scheme and block based AMR to Compressible LES flow solver for IC engines, LES4ICE, 2018 (online), available from <http://www.rs-les4ice.com/Projet/jcms/c_3668480/fr/online-publications>, (accessed on 1 February, 2020).

NASA, Turbulence Modeling Resource (online), available from <https://turbmodels.larc.nasa.gov/>, (accessed on 1 February, 2020).

NASA, CFL3D (online), available from <https://cfl3d.larc.nasa.gov/>, (accessed on 1 February, 2020).

Peskin, C. S., Flow patterns around heart valves: A numerical method, Journal of Computational Physics, Vol.10, No.2 (1972), pp.252-271.

Thobois, L., Rymer, G., Soulères, T. and Poinset, T., Large-eddy simulations in IC engine geometries, SAE Technical Paper (2004). Paper No. 2004-01-1854.

van Leer, B., Towards the ultimate conservative difference scheme V. A second-order sequel to Godunov’s method, Journal of Computational Physics, Vol.32 (1979), pp.101-136.

van Albada, G. D, van Leer, B. and Roberts, W. W., A comparative study of computational methods in cosmic gas dynamics, Astronomy and Astrophysics, Vol.108(1982), pp.76-84.

Weiss, J. M. and Smith, W. A., Preconditioning applied to variable and constant density flows, American Institute of Aeronautics and Astronautics Journal, Vol.33, No.11(1995), pp.2050-2057.

Westbrook, C. K., Mizobuchi, Y., Poinset, T. J., Smith P. J. and Warnatz, J., Computational combustion, Proceedings of the Combustion Institute Vol. 30 (2005), pp. 125-157.

Yakhot, V. and Orszag, S. A., Renormalization group analysis of turbulence: I. Basic Theory, Journal of Scientific Computing, Vol.1, No.1 (1986), pp.3-51.