A Non-Cut Cell Immersed Boundary Method for Use in Icing Simulations

Christian M. Sarofeen and Ralph W. Noack
The Pennsylvania State University, University Park, Pennsylvania

Richard E. Kreeger
Glenn Research Center, Cleveland, Ohio

July 2013

NASA STI Program . . . in Profile

Since its founding, NASA has been dedicated to the advancement of aeronautics and space science. The NASA Scientific and Technical Information (STI) program plays a key part in helping NASA maintain this important role.

The NASA STI Program operates under the auspices of the Agency Chief Information Officer. It collects, organizes, provides for archiving, and disseminates NASA's STI. The NASA STI program provides access to the NASA Aeronautics and Space Database and its public interface, the NASA Technical Reports Server, thus providing one of the largest collections of aeronautical and space science STI in the world. Results are published in both non-NASA channels and by NASA in the NASA STI Report Series, which includes the following report types:

- TECHNICAL PUBLICATION. Reports of completed research or a major significant phase of research that present the results of NASA programs and include extensive data or theoretical analysis. Includes compilations of significant scientific and technical data and information deemed to be of continuing reference value. NASA counterpart of peer-reviewed formal professional papers but has less stringent limitations on manuscript length and extent of graphic presentations.

- TECHNICAL MEMORANDUM. Scientific and technical findings that are preliminary or of specialized interest, e.g., quick release reports, working papers, and bibliographies that contain minimal annotation. Does not contain extensive analysis.

- CONTRACTOR REPORT. Scientific and technical findings by NASA-sponsored contractors and grantees.

- CONFERENCE PUBLICATION. Collected papers from scientific and technical conferences, symposia, seminars, or other meetings sponsored or cosponsored by NASA.

- SPECIAL PUBLICATION. Scientific, technical, or historical information from NASA programs, projects, and missions, often concerned with subjects having substantial public interest.

- TECHNICAL TRANSLATION. English-language translations of foreign scientific and technical material pertinent to NASA's mission.

Specialized services also include creating custom thesauri, building customized databases, organizing and publishing research results.

For more information about the NASA STI program, see the following:

- Access the NASA STI program home page at *http://www.sti.nasa.gov*

- E-mail your question to *help@sti.nasa.gov*

- Fax your question to the NASA STI Information Desk at 443–757–5803

- Phone the NASA STI Information Desk at 443–757–5802

- Write to:
  STI Information Desk
  NASA Center for AeroSpace Information
  7115 Standard Drive
  Hanover, MD 21076–1320

NASA/TM—2013-217881

AIAA–2012–1204

A Non-Cut Cell Immersed Boundary Method for Use in Icing Simulations

Christian M. Sarofeen and Ralph W. Noack
The Pennsylvania State University, University Park, Pennsylvania

Richard E. Kreeger
Glenn Research Center, Cleveland, Ohio

National Aeronautics and
Space Administration

Glenn Research Center
Cleveland, Ohio 44135

July 2013

Acknowledgments

Level of Review: This material has been technically reviewed by technical management.

Available from

NASA Center for Aerospace Information
7115 Standard Drive
Hanover, MD 21076–1320

National Technical Information Service
5301 Shawnee Road
Alexandria, VA 22312

Available electronically at http://www.sti.nasa.gov

A Non-Cut Cell Immersed Boundary Method
for Use in Icing Simulations

Christian M. Sarofeen and Ralph W. Noack
The Pennsylvania State University
University Park, Pennsylvania 16804

Richard E. Kreeger
National Aeronautics and Space Administration
Glenn Research Center
Cleveland, Ohio 44135

Abstract

This paper describes a computational fluid dynamic method used for modelling changes in aircraft geometry due to icing. While an aircraft undergoes icing, the accumulated ice results in a geometric alteration of the aerodynamic surfaces. In computational simulations for icing, it is necessary that the corresponding geometric change is taken into consideration. The method used, herein, for the representation of the geometric change due to icing is a non-cut cell Immersed Boundary Method (IBM). Computational cells that are in a body fitted grid of a clean aerodynamic geometry that are inside a predicted ice formation are identified. An IBM is then used to change these cells from being active computational cells to having properties of viscous solid bodies. This method has been implemented in the NASA developed node centered, finite volume computational fluid dynamics code, FUN3D. The presented capability is tested for two-dimensional airfoils including a clean airfoil, an iced airfoil, and an airfoil in harmonic pitching motion about its quarter chord. For these simulations velocity contours, pressure distributions, coefficients of lift, coefficients of drag, and coefficients of pitching moment about the airfoil's quarter chord are computed and used for comparison against experimental results, a higher order panel method code with viscous effects, XFOIL, and the results from FUN3D's original solution process. The results of the IBM simulations show that the accuracy of the IBM compares satisfactorily with the experimental results, XFOIL results, and the results from FUN3D's original solution process.

Nomenclature

AoA	Angle of attack (Degrees)
dS	Differential surface
Δt	Incremental timestep
∂V	Bounding surface of a volume
E	Total energy
γ	Adiabatic index for air
$\hat{n}$	Surface normal vector
P	Pressure
ρ	Density

R	Residual
τ	Stress tensor
T_x, T_y, T_z	Temperature gradient in the x, y, z coordinate directions
u, v, w	Fluid velocity in the x-axis, y-axis, and z-axis direction respectively
V	Volume
W_x, W_y, W_z	Node velocity in the x-axis, y-axis, and z-axis direction respectively

I. Introduction

Super-cooled water droplets can form in the atmosphere when liquid water quickly moves from air that has a temperature above the freezing point of water to air that has a temperature below the freezing point of water. Although the temperature of these droplets is below freezing they remain as liquid. When these super-cooled liquid water droplets impinge on cold aerodynamic surfaces they freeze into ice. The resulting ice accumulation can adversely impact aircraft performance by significantly reducing lift and increasing drag. Aircraft accidents associated with icing has driven aircraft icing research including computational methods for its simulation.

Traditionally, computational icing analysis has focused on fixed-wing aircrafts, for example LEWICE,[1] LEWICE3D[2] and FENSAP-ICE.[3] Rotorcrafts, on the other hand, include additional complexities including unsteady flow, highly separated flow, wake-fuselage interactions, reversed flow, and stalled airfoil flow. These added complexities create a need for more advanced computational methods for icing. One attempt to include capabilities for rotorcraft icing simulations has been under-way by Kinzel et al.[4]

The method by Kinzel et al[4] uses an unstructured finite-volume CFD approach. This method follows the same general process as the majority of icing codes. First a carrier flow solution (the air flow) is calculated. This carrier flow solution is used to convect the super-cooled water droplets, which either convect around the aerodynamic surfaces or impinge upon them. The solution of the droplet flow field and the air flow field at the aerodynamic surface is used as input for the surface solver. Using an energy and mass balance, the surface solver computes the quantity of water that is convected downstream on the aerodynamic surface by a thin water film and the quantity of water that freezes at each location of the aerodynamic surface resulting in localized accumulation of ice. After some ice accumulation has formed the resulting geometry change of the aerodynamic surface must be considered throughout this process. More information on advanced computational methods for icing can be found in Kinzel et al.[4]

One difficulty that is not addressed by Kinzel et al.[4] is how to take into consideration the geometry change in the icing solvers. Automatic griding could be done for each iteration of a geometric update particularly for two-dimensional analysis. The complex three-dimensional ice shapes that can be found on fixed and rotary wing aircraft can pose significant challenges to even unstructured grid generation methods. Deforming a grid with fixed connectivity to conform to the ice shape is a possible approach. As the ice shape grows and becomes more complex, as illustrated in the ice shape progression shown in Figure 2, the grid quality can significantly suffer especially for three-dimensional ice shapes. The method investigated in this work to accommodate the change in shape due to ice accretion is to use an Immersed Boundary Method (IBM) to represent the comples ice shape with deforming or modifying the grid.
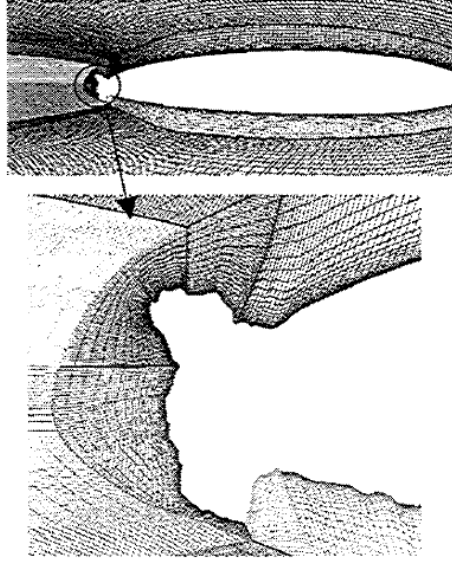
Figure 1: Example of manual meshing of a complex ice shape.[5]

II.   Background

A.   Governing Equations

An Immersed Boundary Method (IBM) was implemented in the NASA FUN3D flow solver. FUN3D uses node centered finite volume cells with second order spatial discretization and varying levels of temporal discretization from first order to third order. The governing equations used by FUN3D are described by Biedron[7] in an inertial reference frame for a moving or static control volume as,

$$\frac{\partial(QV)}{\partial t} + \oint_{\partial V} \left(\overline{\overline{F^*}} - \overline{\overline{F}}_v \right) \cdot \hat{n} \; dS = 0, \tag{1}$$

where $\overline{\overline{F^*}}$ contains the convective fluxes, and $\overline{\overline{F}}_v$ contains the diffusive fluxes. Q is a volume average of the conserved variables, which is defined as,

$$Q = \frac{\int_V q dV}{V}, \tag{2}$$

where the vector of conserved variables q is,

$$q = \begin{bmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ E \end{bmatrix}. \tag{3}$$
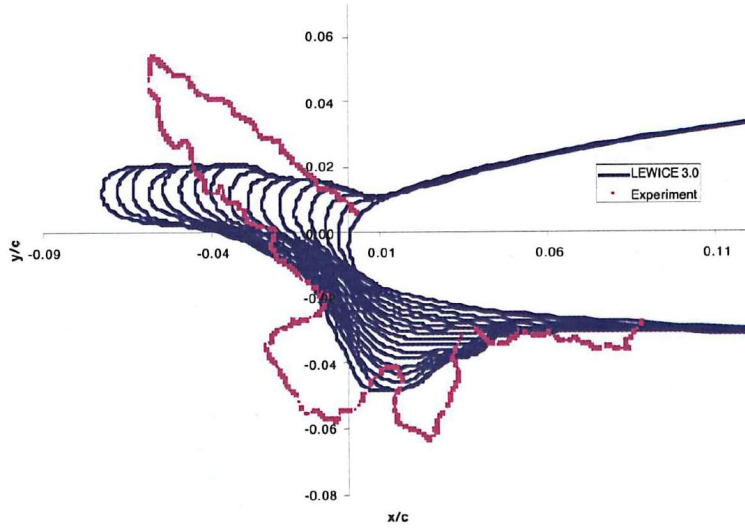
Figure 2: Ice shape geometry change through a LEWICE 3.0 simulation.[6]

The convective and diffusive fluxes are, respectively,

$$
\overline{\overline{F}}^* = \begin{bmatrix} \rho(u - W_x) \\ \rho u(u - W_x) + P \\ \rho v(u - W_x) \\ \rho w(u - W_x) \\ (E + P)(u - W_x) + W_x P \end{bmatrix} \hat{i} + \begin{bmatrix} \rho(v - W_y) \\ \rho u(v - W_y) \\ \rho v(v - W_y) + P \\ \rho w(v - W_y) \\ (E + P)(v - W_y) + W_y P \end{bmatrix} \hat{j}
$$

$$
+ \begin{bmatrix} \rho(w - W_z) \\ \rho u(w - W_z) \\ \rho v(w - W_z) \\ \rho w(w - W_z) + P \\ (E + P)(w - W_z) + W_z P \end{bmatrix} \hat{k} \tag{4}
$$

and

$$
\overline{\overline{F}}_v = \begin{bmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{zx} \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} - kT_x \end{bmatrix} \hat{i} + \begin{bmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{zy} \\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - kT_y \end{bmatrix} \hat{j}
$$

$$
+ \begin{bmatrix} 0 \\ \tau_{xz} \\ \tau_{yz} \\ \tau_{zz} \\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - kT_z \end{bmatrix} \hat{k}. \tag{5}
$$

W_x, W_y and W_z are the control volume face velocities in the x, y and z directions respectively. Definitions for the shear stress terms (τ_{xx}, τ_{xy}, etc.), the heat conduction term and the laminar viscosity can be found in Anderson et al.[8] For compressible calculations these equations are closed with the equation of state for a perfect gas,

$$P = (\gamma - 1)\left(E - \rho\frac{u^2 + v^2 + w^2}{2}\right). \tag{6}$$

FUN3D utilizes a linearized backward-Euler time-differencing scheme to progress the system of equations for each timestep. The temporal discretization is discussed in detail by Biedron et al.[9] but a short description is also presented here.

III. Methods

The Immersed Boundary Method is used herein as a way to represent ice shapes, which are expected to be too complicated to use grid deformation and would be labor intensive to manually re-grid to accommodate the changing iced geometry. The method used is a non-cut cell approach, which uses either fully immersed or fully non-immersed cells to represent the iced geometry. If a node is at a location inside the ice shape it will have solid, viscous wall boundary conditions. This process starts with node identification wher nodes in the computational domain that are inside the ice shape are identified. The governing equations will be modified to enforce the local grid velocity as the fluid velocity at these nodes. A distance field is calculated from the surface made from connecting the outer-most immersed nodes (refered to as the immersed surface) and nodes with a solid wall boundary condition (as would typically be done for a Meneter's SST turbulence model). The governing equations of turbulence are also modified so that the nodes on the immersed surface have a turbulence boundary condition value of a solid wall. Integral forces are calculated at the immersed surface, which is identified by locating immersed nodes which have neighboring computational nodes, to obtain aerodynamic properties (i.e. lift, drag, moments, etc.).

A. Node Identification for the Immersed Boundary Method

The first step of the implementation of the immersed boundary method (IBM) is identifying the ice shape relative to the computational grid. The ice shape will be provided by the icing solver, which extrudes the nodes of an aerodynamic surface in the normal direction at the local rate of ice formation. Since the ice will not always cover the entire aerodynamic surface, the surface grid corresponding to the ice is closed by including the non-iced aerodynamic surfaces. This closed surface is used to identify the nodes in the computational grid which lay inside the ice shape and are marked as immersed nodes.

The IBM node identification has been done by utilizing SUGGAR++[10] and DiRTlib,[11] which are tools used to provide the overset capability within FUN3D. For the overset process SUGGAR++[10] identifies nodes as being in one of the following four main categories for its node identification: **out** nodes, **fringe** nodes, **orphan** nodes, and **in** nodes. Representation of these node types are done by a integer called the iblank value. This information is stored in a domain connectivity information (DCI) file. The IBM node identification takes advantage of this by comparing two sets of domain connectivity information. The two sets of DCI are created with the same set of computational component grids are constructed as follows:

1) The non-immersed/computational DCI:

This grid is used as the actual computational domain; it consists of the farfield grid and the near body grid with nodes inside the body. The iblank values used by FUN3D, in this grid, can contain **in** nodes, **orphan** nodes, **fringe** nodes, and **out** nodes as a typical overset grids does. However there must be a set of nodes inside the immersed body that are all **in** nodes, which will be used for computation with the immersed boundary method. Note: This grid and the corresponding DCI file will be used by FUN3D. (An example of a computational grid is displayed in Figure 3.)
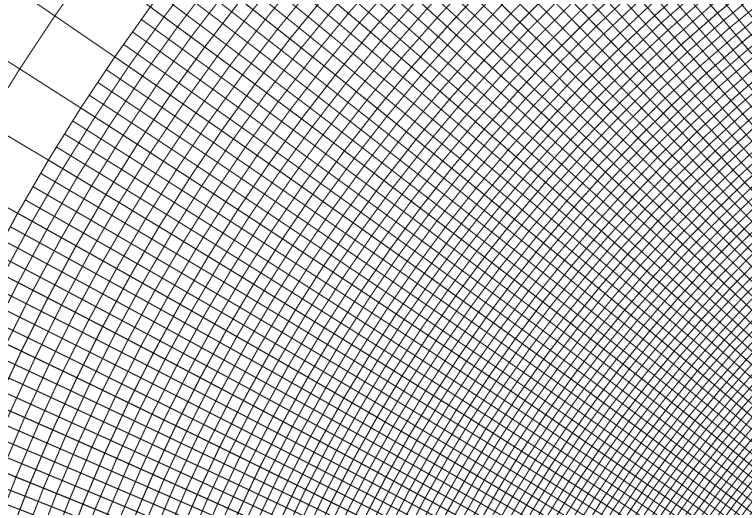
Figure 3: An example of a portion of a two-dimensional computational grid. All nodes in this computational grid are **in** nodes.

2) The "immersed point" DCI:

The immersed point DCI is created with the same set of grids as the non-immersed DCI. The only difference is that the immersed point DCI will flag the immersed boundary nodes, which are all nodes inside the body, as **out** nodes. Suggar++ marks these nodes with the use of a cutter grid. The cutter grid is a volume grid, where one of its boundary surfaces is the ice shape of interest (an example of a cutter grid is shown in Figure 4). By using this cutter grid Suggar++ will mark all nodes inside the ice shape surface with an iblank value signifying that they are now **out** nodes (Figure 5 shows an example of the cut grid with its corresponding **out** nodes). The rest of the nodes will have the same iblank value since they are the same grids previously used. Both DCI files and corresponding grids will have the same nodes, since the added grid is used only for the purpose of defining the ice shape.
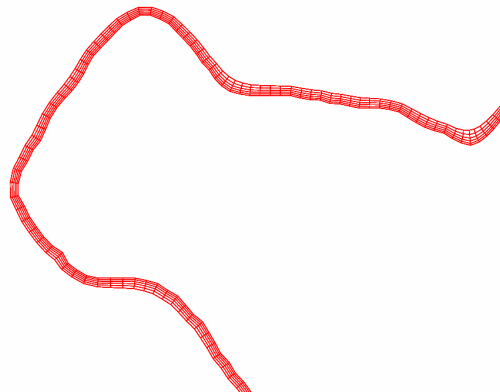
Figure 4: A cutter grid: The grid itself is a volume grid but only the interior boundary is used for cutting.
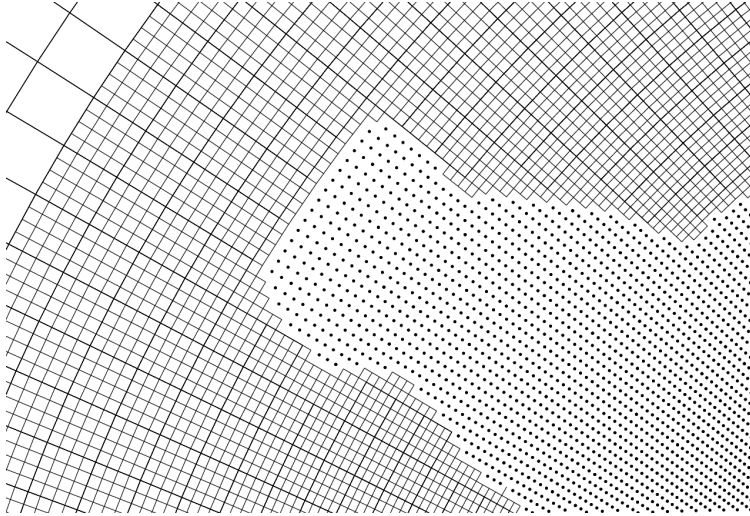
Figure 5: A cut computational grid: The symbols are **out** nodes.

FUN3D was altered to first read the immersed point DCI file, loading the iblank values for each corresponding node. This list of the iblank values are saved and the computational DCI file is loaded. The non-immersed/computational DCI file is loaded second because the computations are run on this set of DCI. By creation both grids have the same nodes and iblank values except that the immersed point grid has the immersed boundary nodes marked as **out** and the computatinoal grid has them marked as **in**. The copy of the immersed point DCI is compared to the data received from the computational grid. Any nodes that are **in** nodes for the non-immersed/computational DCI and **out** nodes for the immersed point DCI are flagged with a new iblank value to signify that it is an immersed node. The nodes marked as immersed are the nodes in the computational grid that are inside the surface grid corresponding to the ice shape.

B.   Governing Equations

The flow solution at the immersed nodes will use a set of modified governing equations. The effect is to enforce a solid-like condition at the immersed nodes in the momentum equations. Applying similar conditions as Cho et al.,[12] Equation 1 is modified to

$$\frac{\partial(Q_V)}{\partial t} + \oint_{\partial V} \left(\overline{\overline{F}}^{i*} - \overline{\overline{F}}_{iv} \right) \cdot \hat{n} \ dS = \frac{\partial(Q_i V)}{\partial t} \tag{7}$$

for the immersed nodes. Where $\overline{\overline{\overline{F}}}^{i*}$, $\overline{\overline{F}}_{iv}$, and Q_i are defined as,

$$
\overline{\overline{\overline{F}}}^{i*} = \begin{bmatrix} \rho(u - W_x) \\ 0 \\ 0 \\ 0 \\ (E + P)(u - W_x) + W_x P \end{bmatrix} \hat{i} + \begin{bmatrix} \rho(v - W_y) \\ 0 \\ 0 \\ 0 \\ (E + P)(v - W_y) + W_y P \end{bmatrix} \hat{j}
$$

$$
+ \begin{bmatrix} \rho(w - W_z) \\ 0 \\ 0 \\ 0 \\ (E + P)(w - W_z) + W_z P \end{bmatrix} \hat{k}, \tag{8}
$$

$$
\overline{\overline{F}}_{iv} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ u\tau_{xx} + v\tau_{xy} + w\tau_{xz} - kT_x \end{bmatrix} \hat{i} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - kT_y \end{bmatrix} \hat{j}
$$

$$
+ \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ u\tau_{yx} + v\tau_{yy} + w\tau_{yz} - kT_x \end{bmatrix} \hat{k}, \tag{9}
$$

and,

$$
Q_i = \frac{\int_V q_i dV}{V}, \tag{10}
$$

and with q_i defined as,

$$
q_i = \begin{bmatrix} 0 \\ \rho\left(\frac{\bar{u}^{n+1} - u^n}{\Delta t}\right) \\ \rho\left(\frac{\bar{v}^{n+1} - v^n}{\Delta t}\right) \\ \rho\left(\frac{\bar{w}^{n+1} - w^n}{\Delta t}\right) \\ 0 \end{bmatrix}. \tag{11}
$$

Here $\bar{u}^{n+1}$, $\bar{v}^{n+1}$, and $\bar{w}^{n+1}$, are the components of the grid velocity for each immersed node at the next time step, and u^n, v^n, and w^n are the components of the fluid velocity at the node.

In contrast to the approach of Cho et al.,[12] a source term is not actually being added to the equations of motion. The equations of motion are shown in control volume form for consistency with Equation 1. The implications of these modified governing equations, for incompressible flow, are the following: the velocity at the immersed nodes will be zero for static cases and the grid velocity for moving grid cases, and the pressure

gradient with respect to the normal of the interface of the immersed nodes and the rest of the computational domain will be zero in the vicinity of the interface.

The first implication is trivial, using first order differencing in time for simplicity and taking the three momentum terms from Equation 7, including Equation 8, Equation 9, Equation 10, and Equation 11. Inside the immersed body the three velocity components and their spacial derivatives are zero. The equations of motion simplify to

$$
\frac{1}{V} \int_V \begin{bmatrix} \rho \frac{u^{n+1}-u^n}{\Delta t} \\ \rho \frac{v^{n+1}-v^n}{\Delta t} \\ \rho \frac{w^{n+1}-w^n}{\Delta t} \end{bmatrix} dV = \frac{1}{V} \int_V \begin{bmatrix} \rho \left(\frac{u_i^{n+1}-u^n}{\Delta t} \right) \\ \rho \left(\frac{v_i^{n+1}-v^n}{\Delta t} \right) \\ \rho \left(\frac{w_i^{n+1}-w^n}{\Delta t} \right) \end{bmatrix} dV.
\tag{12}
$$

Here the subscript i denotes the velocity of the grid at the immersed boundary node. Enforcement of the pressure boundary condition at the immersed surface comes from the momentum equations when approaching the surface of the immersed boundary nodes from the computational domain along the normal to the surface.

For compressible flow the implications are more complicated. This is due to two things. First, the conservation of mass does not simplify to the partial derivative of velocity in the y direction with respect to the y direction equalling zero. This is because the partial derivative of density with respect to time and the spatial coordinates are not necessarily zero. The second complication is that there is an added equation for the conservation of energy compared to incompressible flows. At a solid wall the momentum and mass conservation equations do not simplify to the derivative of pressure in the normal direction equalling zero as the incompressible case does. To represent a particular solid using this method the corresponding sink or source terms must be added to the energy equation. For the method presented here the energy equation is left in its original form for the immersed body. The immersed body is still simulated as a solid body since the velocity of the immersed nodes are forced to be the local grid velocity. Although, the thermodynamic properties of the body will be that of static air, which is not convecting even though there can be a temperature gradient, since the energy equation is left in its original form. In contrast, conduction is calculated throughout the body with conductivity of air since the energy equation is in its original form.

The actual implementation into FUN3D differs slightly from the governing equations previously discussed. In the FUN3D algorithm, as the solver goes through the time incrementation loops, the ρu, ρv, and ρw are set to the grid velocity at the time step $n+1$ multiplied by the local density for immersed nodes. This in effect sets the velocity as a boundary condition would. However, if the solver was allowed to continue through its normal solution scheme these values would simply be overwritten by a new solution value. Therefore the momentum equations at the immersed nodes are changed to specified value equations.

C. Turbulence Model

The last consideration for the immersed boundary method is to handle the turbulence equations in the immersed body. The turbulence model used herein is Menter's shear stress transport (SST) k-ω turbulence model.

1. Governing Equations

The turbulence model used (Menter's shear stress transport k-ω model) is consistent with Menter.[13] This is a two-equation model that can be written in conservative form as

$$
\frac{\partial \rho k}{\partial t} + \frac{\partial \rho u_j k}{\partial x_j} = P_k - \beta^* \rho \omega k + \frac{\partial}{\partial x_j} \left[(\mu + \sigma_{k1} \mu_t) \frac{\partial k}{\partial x_j} \right],
\tag{13}
$$

and

$$
\frac{\partial \rho \omega}{\partial t} + \frac{\partial \rho u_j \omega}{\partial x_j} = \gamma_1 P_\omega - \beta_1 \rho \omega^2 + \frac{\partial}{\partial x_j} \left[(\mu + \sigma_{\omega 1} \mu_t) \frac{\partial \omega}{\partial x_j} \right].
\tag{14}
$$

Where,

$$P_k = \mu_t \frac{\partial u_i}{\partial x_j} \left(frac\partial u_i \partial x_j + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij} \frac{\partial u_i}{\partial x_j}, \tag{15}$$

and

$$P_\omega = \rho \frac{\partial u_i}{\partial x_j} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \rho \omega \delta_{ij} \frac{\partial u_i}{\partial x_j}. \tag{16}$$

As suggested by Cho et al.[12] ω is set to the freestream value and k is set to zero inside the immersed body. The value for ω at a wall is set consistently with FUN3D's boundary conditions by use of their functions, which is defined as,

$$\omega_{wall} = \frac{600\nu}{\beta_1 (\Delta d_1)^2}. \tag{17}$$

Where in FUN3D Δd_1 is defined as the average distance from a wall node to its neighboring nodes.

2. Isolation of the Immersed Surface

The immersed boundary surface between the flow/non-immersed and immersed nodes is created by connecting the immersed nodes that have neighboring flow/non-immersed nodes, must be isolated for two reasons. The first reason is that this surface will be used by FUN3D's boundary surface force calculation routines, and the second is that the surface needs to be used by FUN3D's methods to calculate the distance field that is used by Menter's SST turbulence model. The distance field is a scalar field composed of the distance between every node and the closest node or face of any solid boundary. After the distance for the computational nodes are obtained, the nodes on the solid boundaries are given a distance value that is the average distance between the solid boundary node and the neighboring nodes (boundary or computational). The distance value of the solid boundary nodes are used as Δd_1 for the turbulence boundary condition on ω_{wall}.

The algorithm used in this effort for finding the immersed surface is a cell searching algorithm. The algorithm first looks for cells that are on the immersed surface. These cells will have at least one node marked as immersed and at least one node outside the immersed body. When a cell is found that meets this criterion the algorithm selects faces that only have nodes that are marked as immersed. These cell faces will be the faces of the immersed surface. An example of an immersed body simulation representing an ice shape on the leading edge of an airfoil is shown in Figure 6 and Figure 7.
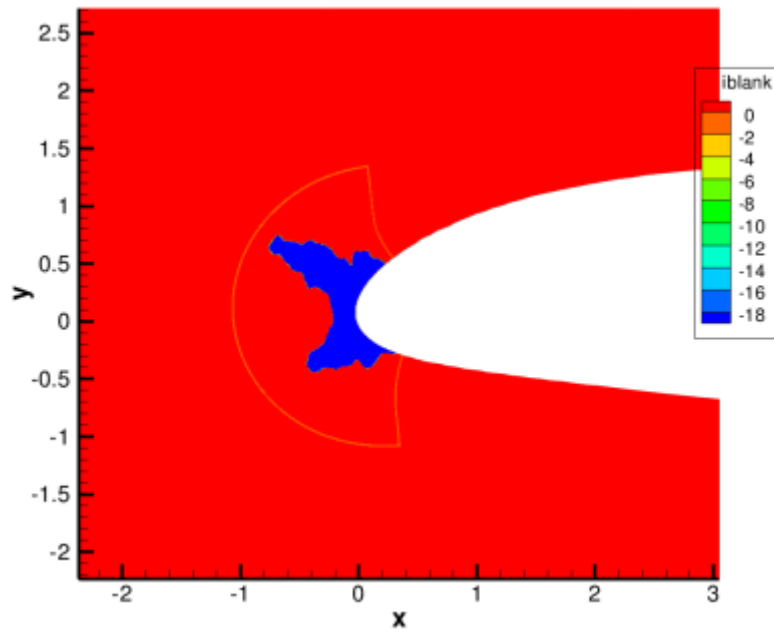
Figure 6: An example of an ice shape represented as an immersed body. The immersed body is in blue while the rest of the computational domain is in red. Both are represented as an integer value of -20 and 1 respectively.
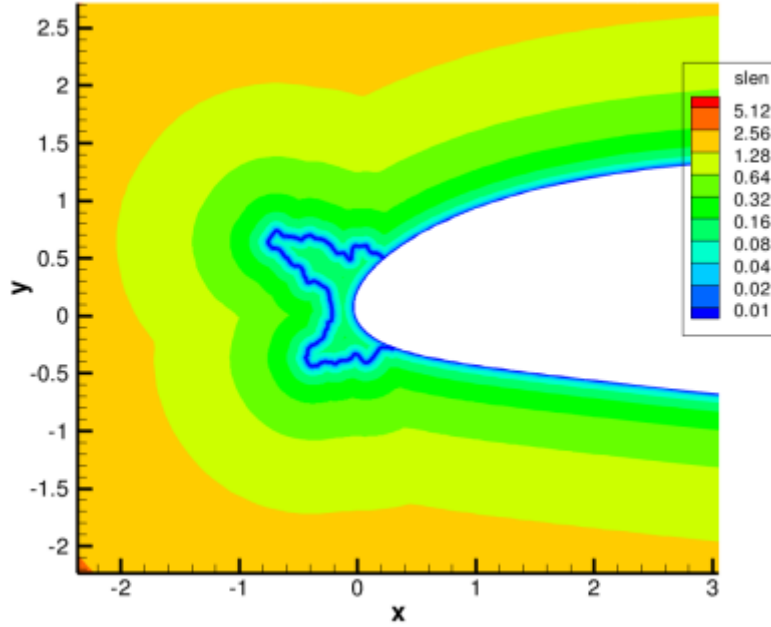
Figure 7: Color contour plot representing minimum distance from the closest solid boundary, corresponds to Figure 6.

D. Calculations of the Aerodynamic Forces

Traditionally the aerodynamic forces are calculated from the fluid stress and pressure on the surface of a boundary. Since the immersed body will also represent a solid surface the forces acting on the immersed body must be taken into consideration. Kim et al.[14] propose to do this by integrating the added source term to the momenutm equations, which represents a body force, at the immersed cells, required to bring the fluid to a particular velocity (zero velocity for static simulations). Inside the immersed body this is the equivalent of the volume integration of three terms, the pressure term in the momentum equation,

$$\frac{\partial P \delta_{ij}}{\partial x_j},\tag{18}$$

the stress term,

$$-\frac{\partial \tau_{ij}}{\partial x_j},\tag{19}$$

and the remaining term in Kim et al.[14] and Cho et al.,[12] the inertial term,

$$\frac{\rho(u_{bi}^{n+1} - u_i^n)}{\Delta t}.\tag{20}$$

The sum of these three terms are equal to the negative residual mentioned in Biedron[7] plus the inertial term (Equation 20). The inertial term is not a standard aerodynamic force and should not be taken into consideration. This inertial force is necessary to enforce acceleration of the immersed body but it is not felt by the aerodynamic surface. Using this method for node centered solvers has a larger source of error

than the inertial term. Nodes of an unstructured grid and their vertices are shown in Figure 8a with the corresponding control volume for node one shown in Figure 8b. As seen in these figures the control volume of the nodes do not align with their vertices. Since the residual is associated with the control volume, but the wall boundary condition is associated with the node, integration of the residual as mentioned does not correspond with the forces on a wall. In fact, integration of the residuals will calculate the aerodynamic force at some distance from the wall between the wall node, and the first node normal to the wall.

(a) Unstructured grid nodes and connecting vertices, line intersections representing the node locations.

(b) The dual grid control volume for node one. The grey lines are the boundaries of the control volume about node one.
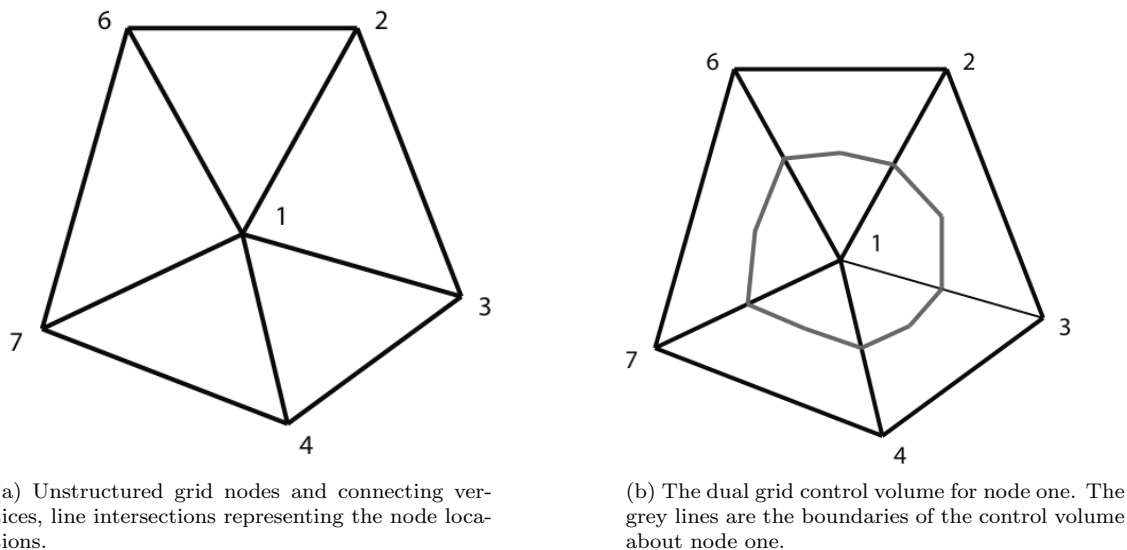
Figure 8: Sample of an unstructured grid and its corresponding dual grid.

For this reason the the aerodynamic force is calculated from the stress and the pressure on the immersed surface by use of

$$F = \int_s (P\delta_{ij} - \tau_{ij}) \cdot \hat{n} ds. \tag{21}$$

It is useful to calculate the total forces of the immersed boundary surface and the aerodynamic body together. For this the integral of Equation 21 is added to the aerodynamic force of the body. With reference to simulations where the immersed body overlaps some surface of the aerodynamic body, the overlapped surface must be identified and the associated force removed. Removal of this force results in the force along the combined global surface of the aerodynamic body and immersed surface.

IV.   Results

A.   Static 2-D Clean NACA 23012 with a Contour Fitted Grid Using The IBM

1.   Experimental Setup

A NACA 23012 airfoil is used for the two-dimensional static validation of the IBM with simulation parameters to match the experiments performed by Busch et al.[15] The airfoil used in these experiments was an NACA 23012 with a chord of 18 inches and a span of 33.563 inches. The experiment was performed in the University of Illinois subsonic, low-turbulence, open-return wind tunnel with a test section of 2.8 ft in height and 4 ft wide. The turbulence of the wind tunnel used was designed to produce less than 0.1% turbulence intensity in

the test section. Since Busch et al.[15] focused primarily on an iced NACA 23012 and did not include extensive performance data of a clean NACA 23012, validation will be compared against results from XFOIL and from a body fitted simulation with the original FUN3D solution process.

2. Computational Setup

The computational simulations use an NACA 23012 with an 18 inch chord and properties to match experiments by Busch et al.[15] All simulations have a free-stream turbulence intensity of 0.1%, a Reynolds number of 1.8 million and a free-stream Mach number of 0.18. The angle of attack is varied, with individual static simulations, from 0 degrees to 18 degrees. The interior of the airfoil is gridded as shown in Figure 9. The nodes contained in the section of the grid internal to the airfoil walls are set as immersed boundary nodes, which enforces a solid wall boundary condition at the aifoil surface. These nodes are shown as blue dots in Figure 10. Drag, lift, and pitching moment are compared with both XFOIL and the results obtained from a body fitted simulation with FUN3D's original solution process.
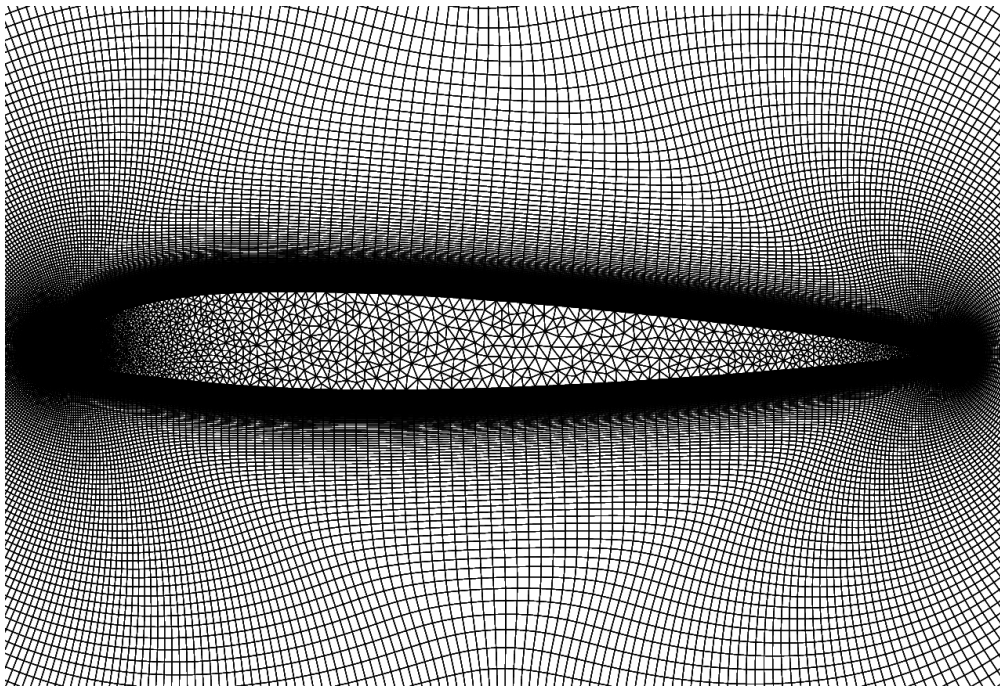
Figure 9: Computational grid for the simulation of a clean NACA 23012, with the Immersed Boundary Method representing the wall of the airfoil.
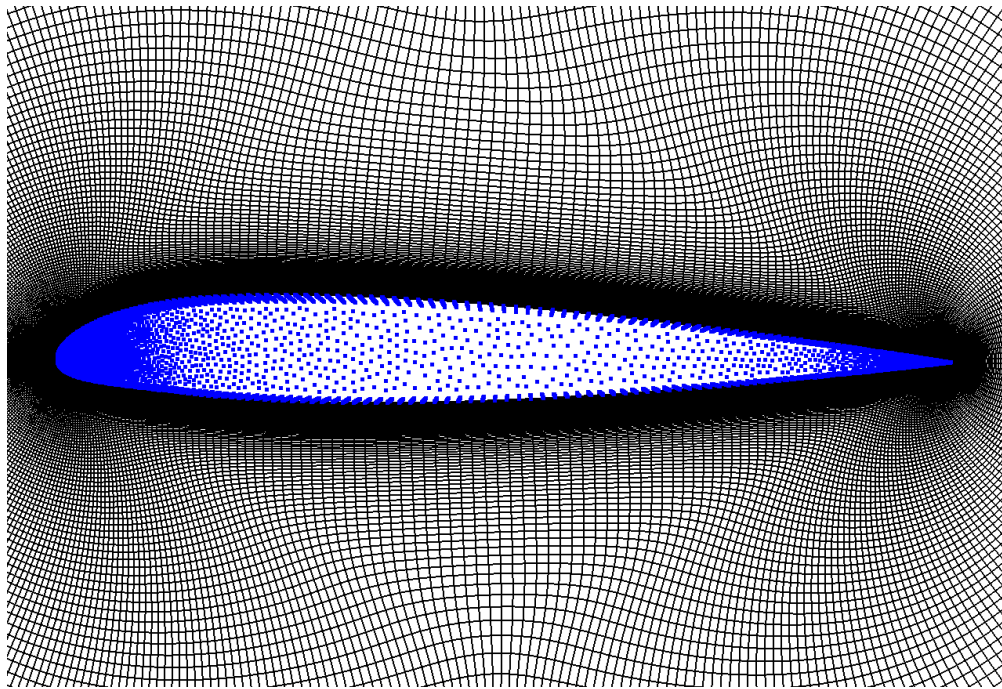
Figure 10: The computational grid shown in figure 9, blue points are immersed boundary nodes.

3. Results

The results are compared with both XFOIL and the results obtained from a body fitted simulation with FUN3D's original solution process. Results from the Immersed Boundary Method are expected to be similar to the results from FUN3D's normal solution process since they are both solved with mostly the same process. Aerodynamic forces are compared in Figure 11. Some discrepancies can be seen in the aerodynamic forces at higher angles of attack but this is attributed to the unsteadiness of these flows when the airfoil is in stall. Pressure and velocity magnitude contours are compared between the IBM simulation in Figure 12. Some discrepancies can be seen in the pressure and velocity magnitude contours at the higher angles of attack, although this, once again, can be attributed to the unsteadiness of these flows indicated by the separated boundary layer at $16°$ angle of attack. Overall the solution of the IBM is very similar to the solution using FUN3D's solid wall viscous boundary condition.
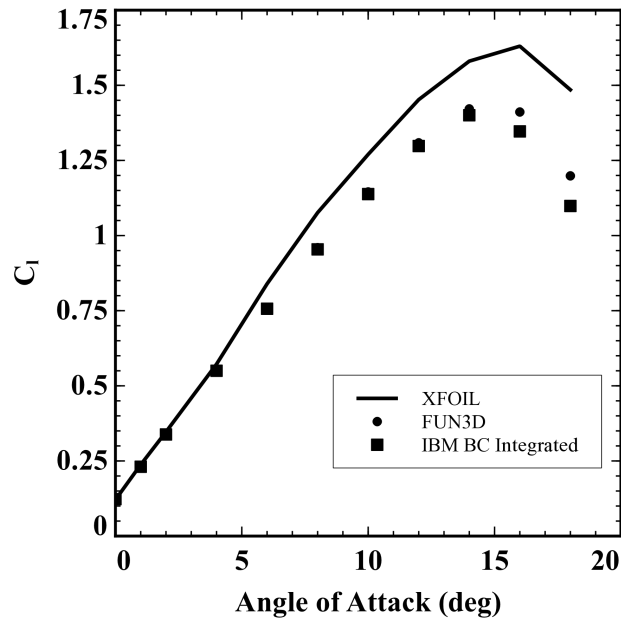
(a) Coefficient of lift
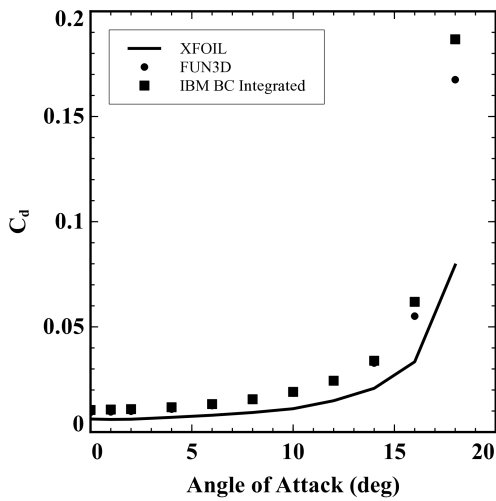
(b) Coefficient of drag
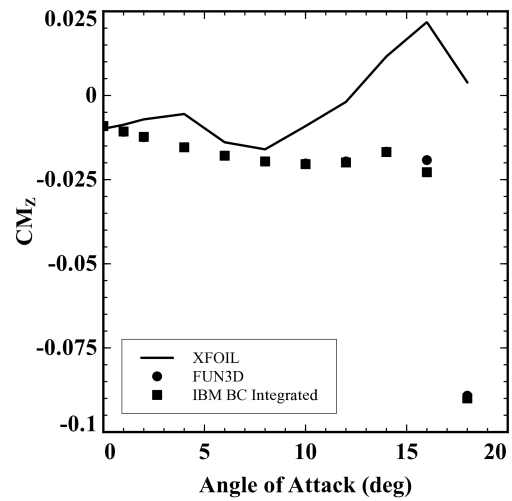
(c) Coefficient of pitching moment

Figure 11: Comparison of aerodynamic forces predicted by the IBM method (squares), the simulation using FUN3D's solid boundary condition (circles), and XFOIL (line) at various angles of attack.
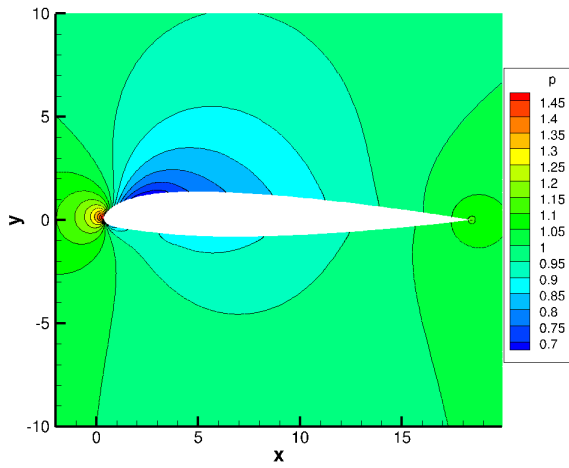
(a) Pressure at 0° angle of attack.
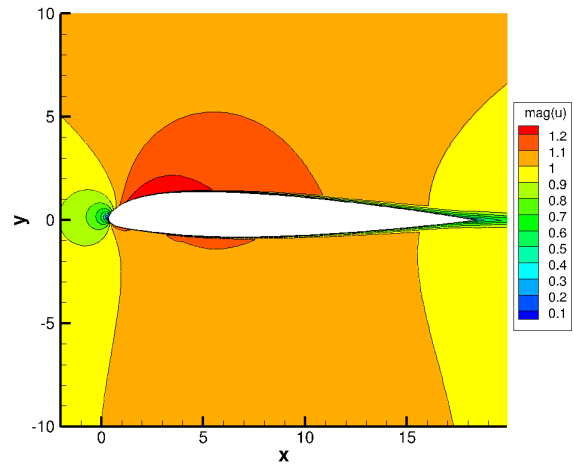
(b) Velocity at 0° angle of attack.
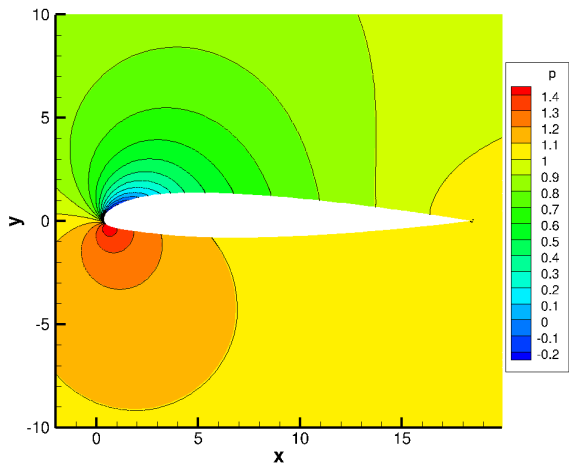
(c) Pressure at 8° angle of attack.
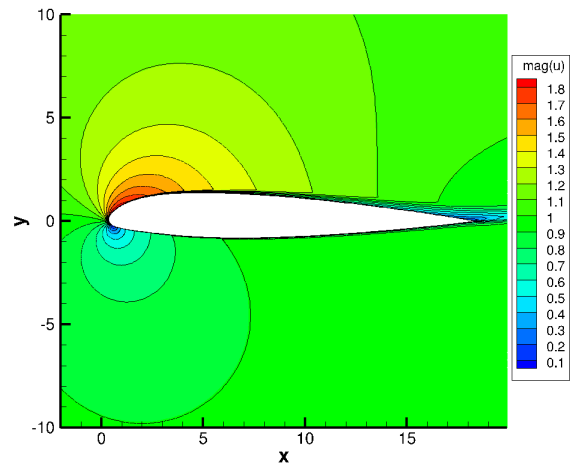
(d) Velocity at 8° angle of attack.
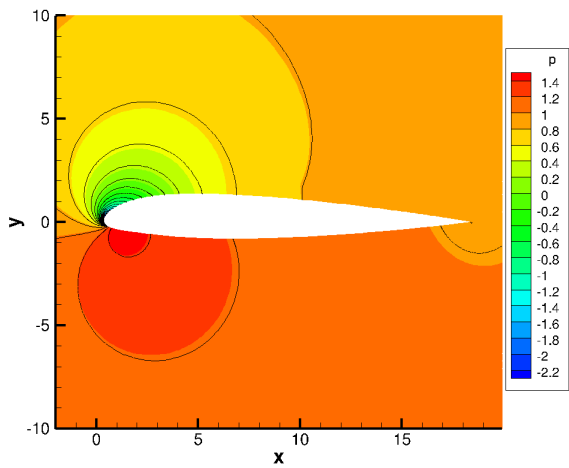
(e) Pressure at 16° angle of attack.
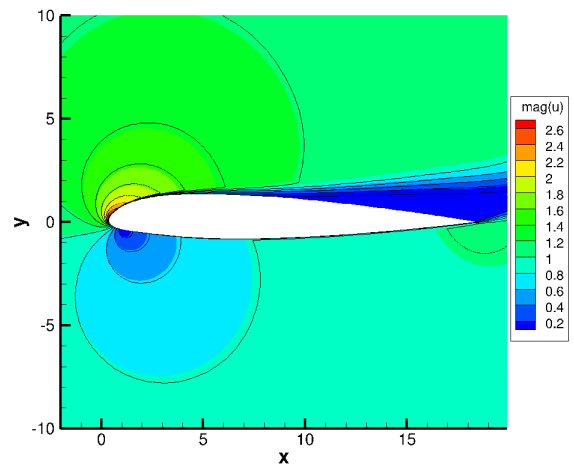
(f) Velocity at 16° angle of attack.

Figure 12: Color contour plots from results of the IBM, and line contour plots, overlain, of the FUN3D results from Section. Pressure and velocity are non-dimensional. (Note: the freestream velocity changes, not the orientation of the body.

B. 2-D Iced NACA 23012 With the Immersed Boundary Method

The next simulation for testing the Immersed Boundary Method is of a two-dimensional, iced NACA 23012 airfoil. Two simulations have been performed and are compared to experiments of Busch et al.[15] The first simulation is a body fitted grid run with a solid, viscous wall boundary condition. This simulation provides a baseline for the accuracy of the aerodynamic forces from a two-dimensional computational simulation of a large ice shape. The second simulation uses the Immersed Boundary Method to model the ice. This simulation uses a clean airfoil fitted grid with the IBM representing the ice.

1. Experimental Setup

The experimental data used for comparison are the experiments from Busch et al.[15] mentioned in Section A. Coefficients of lift and pitching moment were obtained using a three-component force balance and by integration of measured surface pressure around the airfoil. The drag coefficient was computed by collecting pressure data from a traversable wake rake.

2. Computational Setup

The computational simulation with FUN3D and with the use of the IBM uses the same 18 inch chord NACA 23012. Free-stream turbulence is set to 0.1% turbulence intensity to match the wind tunnel used by Busch et al.[15] All simulations are run at a Reynolds number of 1.8 million and a Mach number of 0.18. The angle of attack is varied, with individual static simulations, from 0 degrees to 10 degrees. Drag, lift, and pitching moment are compared against experimental results from Busch et al.[15] A close up image of the ice shape region with the computational grid used for the body fitted FUN3D simulation is shown in Figure 13. A close up image of the ice shape region with the computational grid used for the IBM is shown in Figure 14. The nodes that are inside the ice shape have been removed in this figure. For the Immersed Boundary Method, a body fitted clean airfoil grid is used with an refined structured grid in the vicinity of the ice. The refined grid and clean airfoil grid are joined by use of the overset capability within FUN3D.
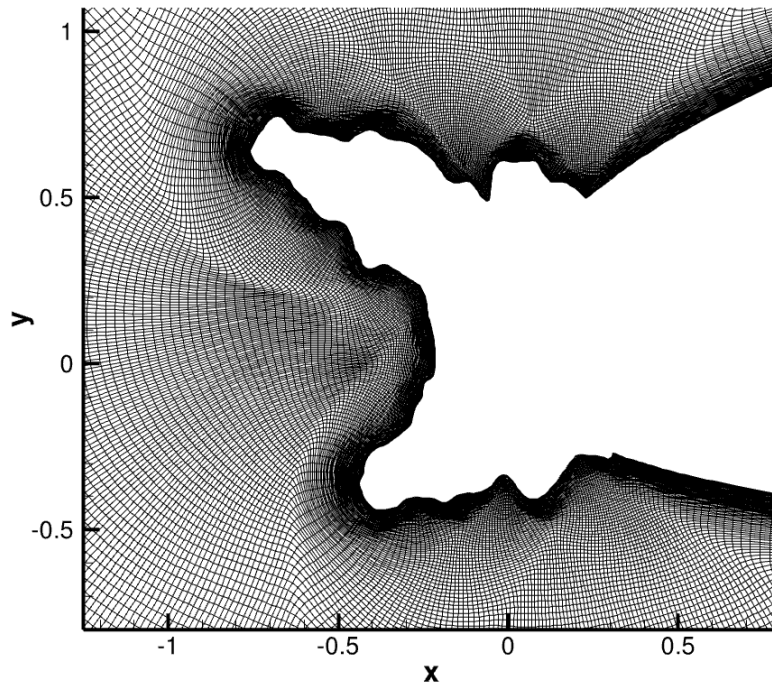
Figure 13: The ice shape region of the computational grid used for the body fitted simulation. Interior is modeled with a solid, viscous wall boundary condition.
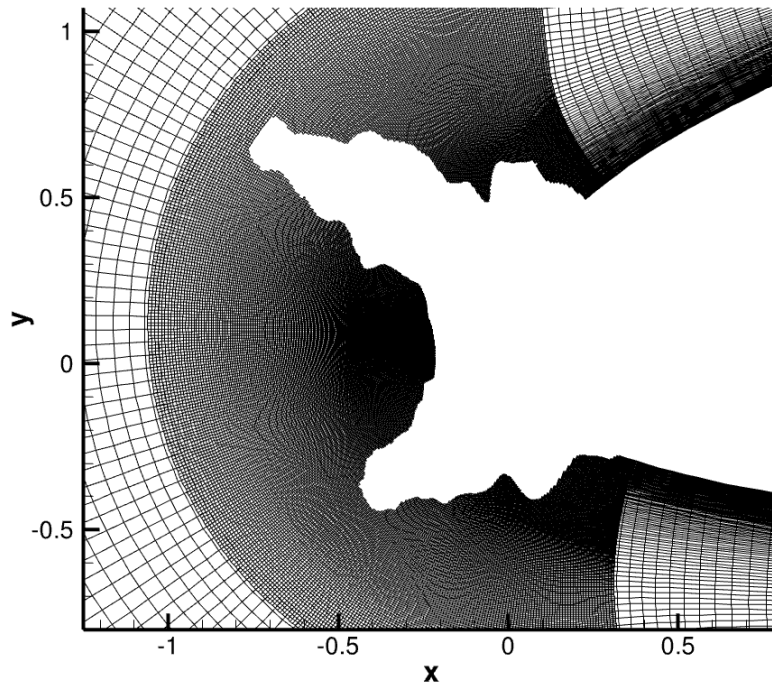
Figure 14: The ice shape region of the computational grid used for the IBM simulation. The interior of the ice shape is simulated with the IBM while the interior of the airfoil uses a solid wall boundary condition.

3. Results

Both the body fitted simulation and the IBM simulation of the iced NACA 23012 are compared with the experimental results of Busch et al.[15] and plotted with the clean NACA 23012 results from XFOIL for reference in Figure 15. Pressure and velocity magnitude contour plots are compared between the IBM simulation and the body fitted grid simulation in Figure 16. Both simulations show the correct trends for its aerodynamic forces. When compared against experimental data the body fitted grid simulation appears more accurate. The main cause of this is most likely due to the front top and front bottom tips of the ice shape. In the body fitted grid these areas maintain clean airfoil boundary layer spacing in the normal direction around the ice, where the IBM simulation has a much larger spacing due to the construction of the iced region. This larger spacing both decreases the shear stress calculated at these locations and causes the turbulence model to produce less accurate turbulence properties which convects downstream in the vicinity of the airfoil. Overall the aerodynamic forces, pressure contour plots, and velocity magnitude contour plots are very similar. There are some discrepancies in the pressure and magnitude of velocity contour plots behind the ice shape between the two simulations although this is attributed to the unsteadiness of these regions.
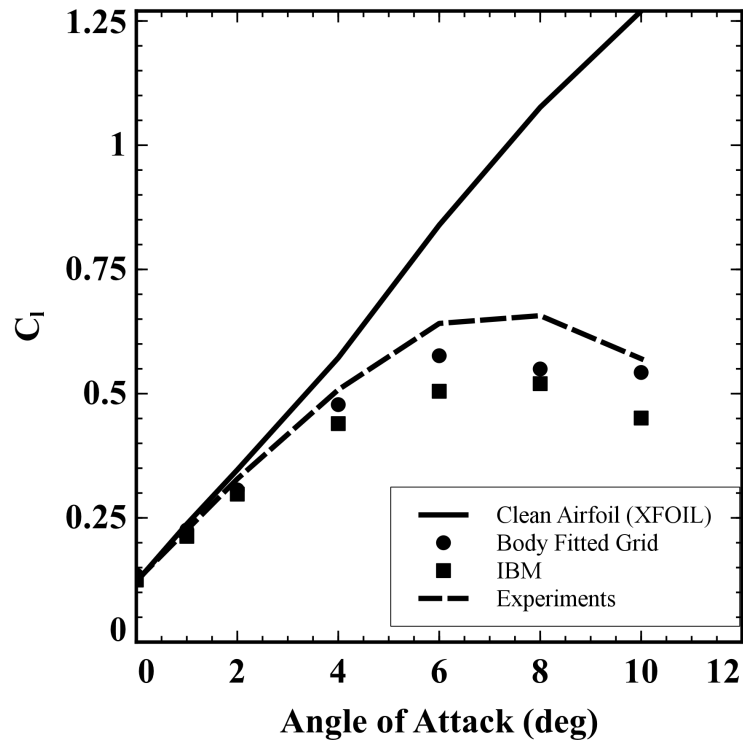
(a) Coefficient of lift versus angle of attack for an iced NACA 23012.
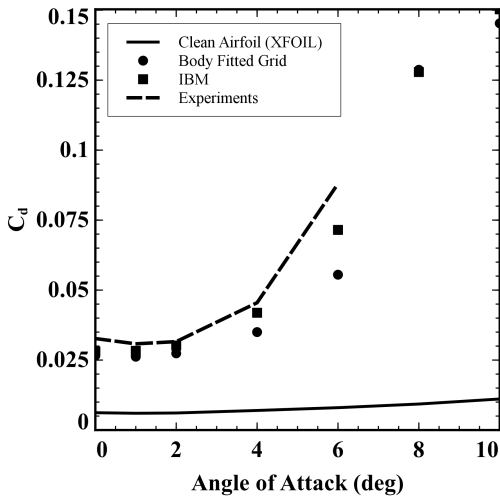
(b) Coefficient of drag versus angle of attack for an iced NACA 23012.
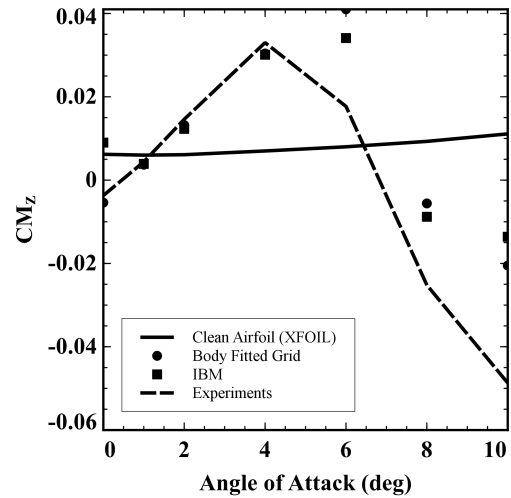
(c) Coefficient of pitching moment around the quarter chord versus angle of attack for an iced NACA 23012.

Figure 15: Comparison of results from the IBM method (squares), results from the body fitted FUN3D simulation(circles), experimental results from Busch et al.[15](dashed line) and clean airfoil results from XFOIL for reference(solid line).
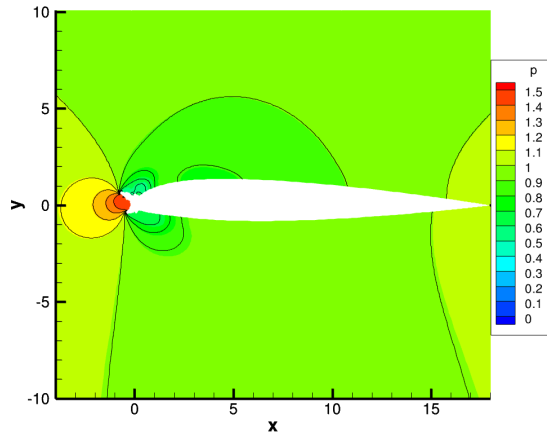
(a) Pressure at 0° angle of attack. IBM data shown with a color contour, while body fitted simulation contour shown with lines.
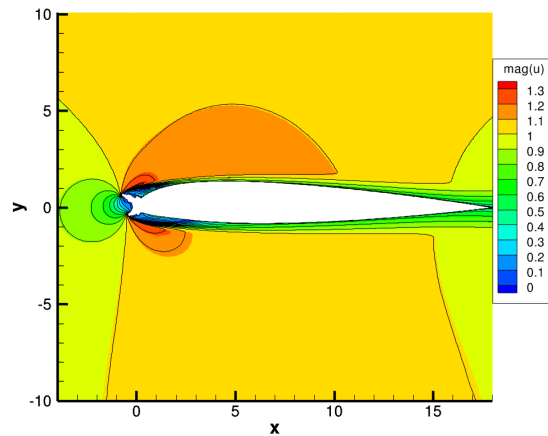
(b) Velocity at 0° angle of attack. IBM data shown with a color contour, while body fitted simulation contour shown with lines.
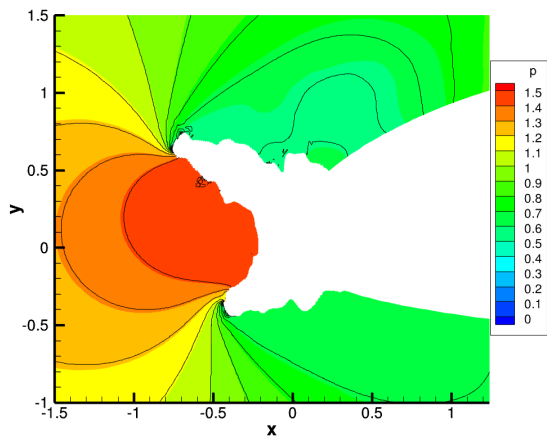
(c) A close up of the ice region. Pressure shown at 0° angle of attack. IBM data shown with a color contour, while body fitted simulation contour shown with lines.
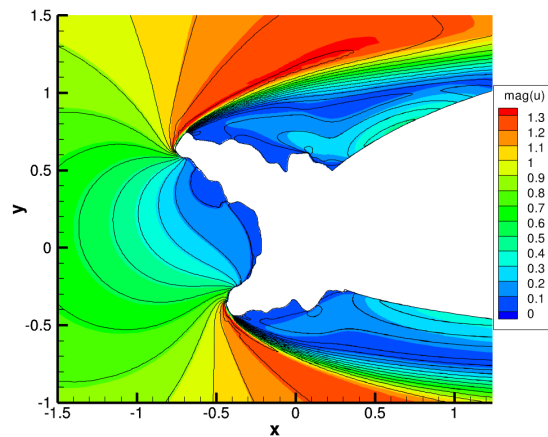
(d) A close up of the ice region. Velocity shown at 0° angle of attack. IBM data shown with a color contour, while body fitted simulation contour shown with lines.

Figure 16: Contour plots of pressure and velocity of an iced NACA 23012. Contour plots of the IBM simulated ice shown in color while contour of the body fitted simulation shown with lines.

C. An Oscillating 2-D NACA 0012

1. *Experimental Setup*

Experimental data used for comparison was taken from Lee and Gerontakos.[16] In this set of experiments an NACA 0012 airfoil was tested at a Reynolds number of 300,000. This airfoil was oscillated about it's quarter chord location at various frequencies and amplitudes. The experiment was conducted in the low-speed, suction-type wind tunnel in the Aerodynamics Laboratory at McGill University. This wind tunnel has a test section of 0.9 m x 1.2 m and a free-stream turbulence intensity of 0.08% at a free-stream velocity of 35 meters per second. The chord of the NACA 0012 airfoil is 0.15m and has a span of 37.5 cm. The airfoil was fitted with two 30 cm diameter stationary end plates with sharp leading edges to help reduce airfoil end effects. The particular case that is used for validation has a frequency of 7.14Hz with a mean angle of attack of 10 degrees and an amplitude of 15 degrees.

Computational parameters are set to match the experiment performed by Lee and Gerontakos.[16] An NACA 0012 clean airfoil is used with a chord of 0.15m. This airfoil is simulated as two-dimensional, at a Reynolds number of 300,000 and a freestream speed of 35 m/s. A body fitted grid is constructed and then the interior of the airfoil is gridded, with structured cells in the vicinity of the airfoil surface, and an interior unstructured grid, as shown in Figure 17. The airfoil is represented with the IBM, with the nodes interior to the airfoil surface marked as immersed nodes, as shown in Figure 19. The simulation uses prescribed rigid grid motion for the harmonic oscillating pitching motion.
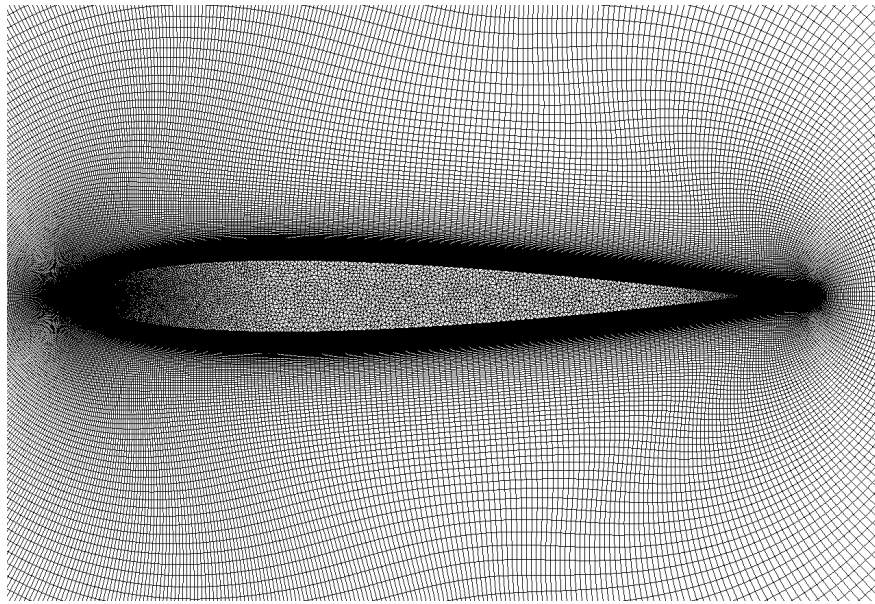
Figure 17: Computational grid for the simulation of a clean NACA 0012, with the Immersed Boundary Method representing the wall of the airfoil.
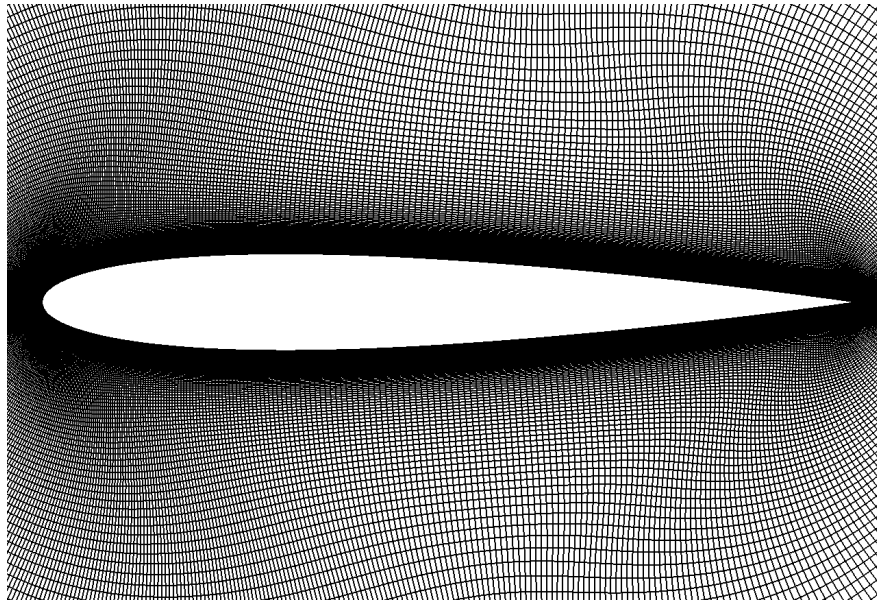
Figure 18: The computational grid shown in Figure 17, with immersed body nodes removed.
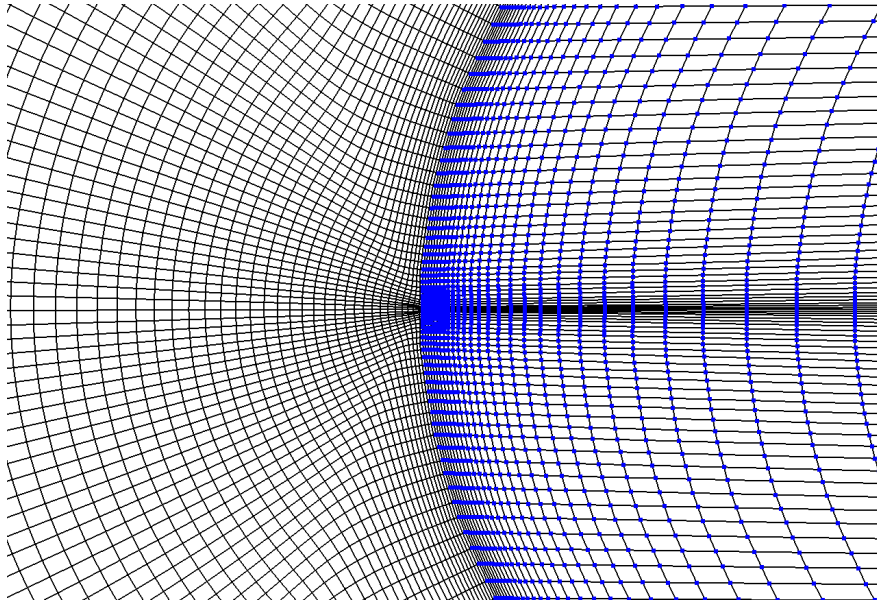
Figure 19: A close up of the leading edge of the computational mesh shown in Figure 17, blue points are immersed boundary nodes.

3. Results

Comparison of the predicted aerodynamic forces are shown in Figures 20 and 21. In both figures the airfoil increases its angle of attack along the top lines (both coefficient of drag and lift), it reaches its maximum angle of attack at 25 degrees and decreases its angle of attack along the bottom lines. The IBM follows

the physics associated with the lift and drag well. The linear lift portion of the curve agrees well with experiments, this portion of the movement is shown in Figures 23c, 23d, 22a, 22b, 25c, 25d, 24a, and 24b. There is a prominent increase in lift at a rate above the linear lift line as it sheds a large vortex at about 22 degrees angle of attack (shown in Figures 22c and 24c), which is in close proximity to the experimental data. There is a large decrease in both lift and drag after the large vortex sheds and moves away from the airfoil when it is very close to 25 degrees angle of attack similar to the experiments (shown in Figures 22d, 24d). Lastly, the airfoil's angle of attack decreases and eventually the boundary layer reattaches (shown in Figures 22e, 22f, 23a, 23b, 24e, 24f, 25a, and 25b). There are discrepancies between the experimental data and the computational simulation, particularly in drag, which is known to be particularly difficult to predict accurately in computational simulations for unsteady motion. This is mainly due to the size of the grid and inaccurate turbulent properties being propagated downstream. Highly separated flow, which occurs during stall, is composed of three-dimensional structures, accounting for the other disagreements between the computational and experimental results.
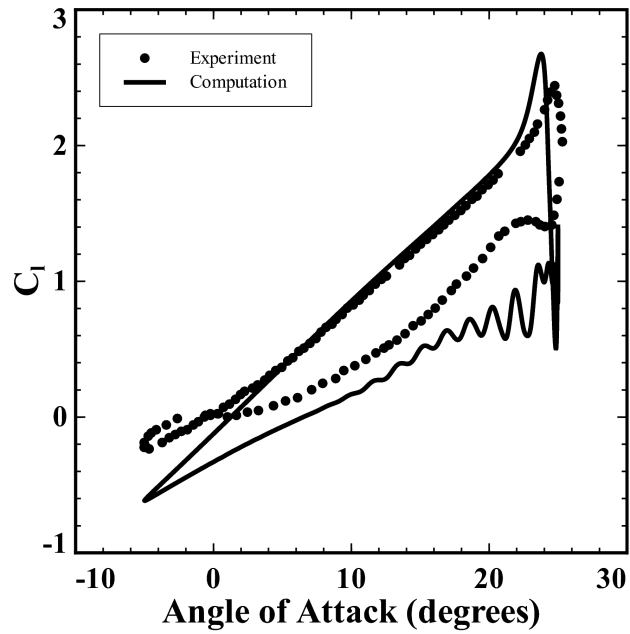
Figure 20: Coefficient of drag versus angle of attack for an oscillating NACA 0012. Experiment shown with dots and simulation results with the IBM with a solid line.
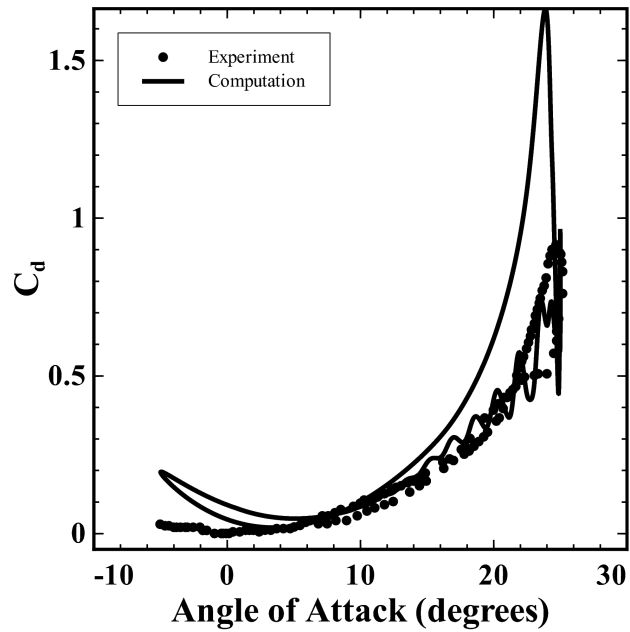
Figure 21: Coefficient of drag versus angle of attack for an oscillating NACA 0012. Experiment shown with dots and simulation results with the IBM with a solid line.
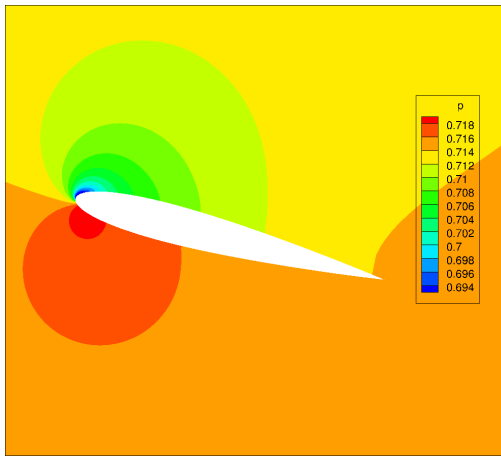
(a) Approximately 10° inc.
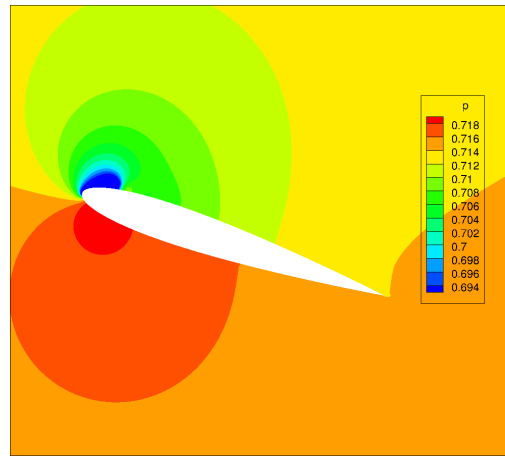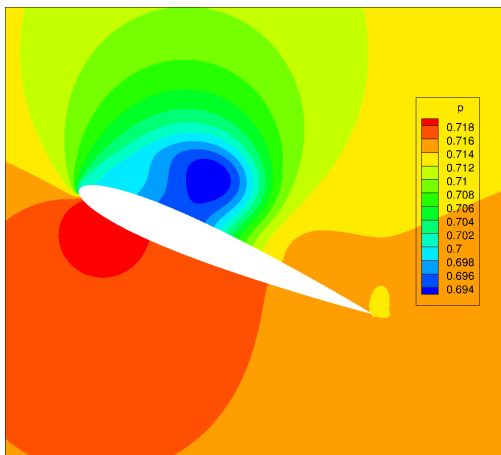
(b) Approximately 15° inc.

(c) Approximately 20° inc.
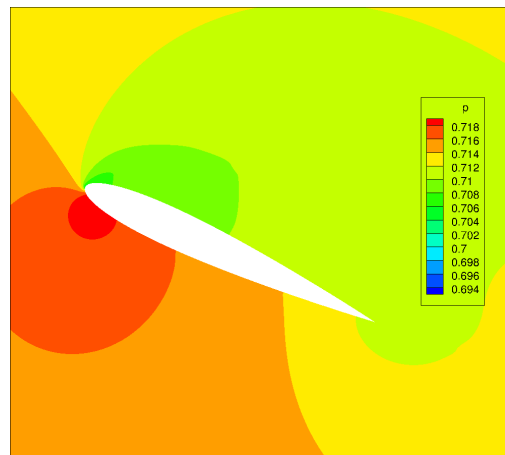
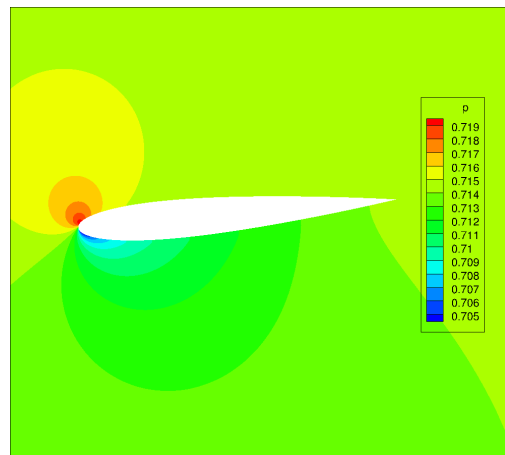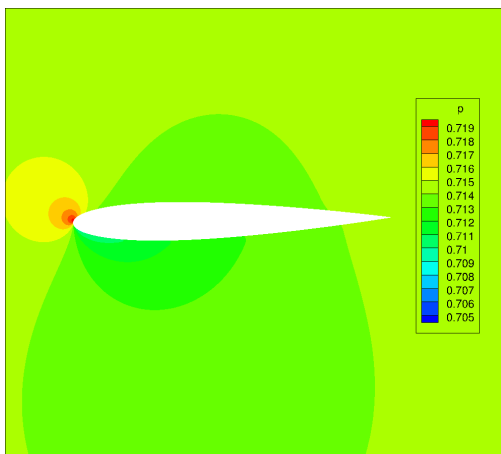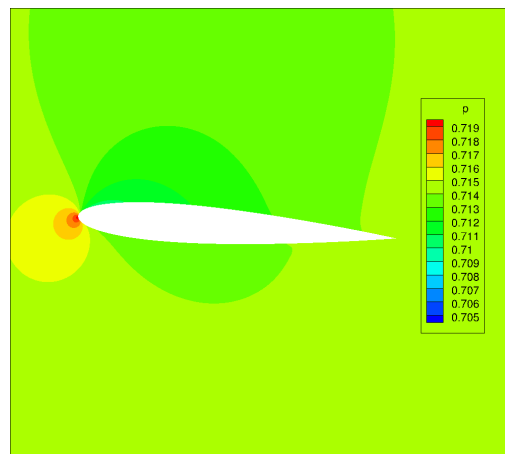(d) Approximately 25° inc.
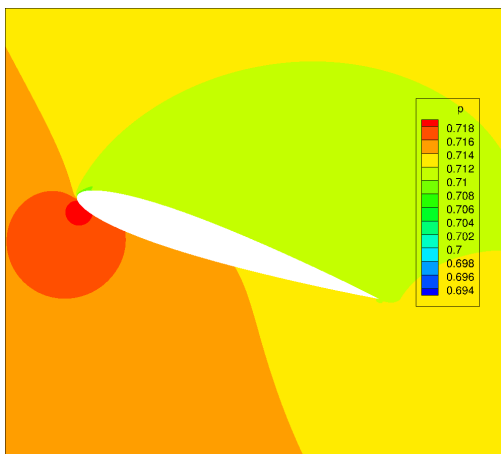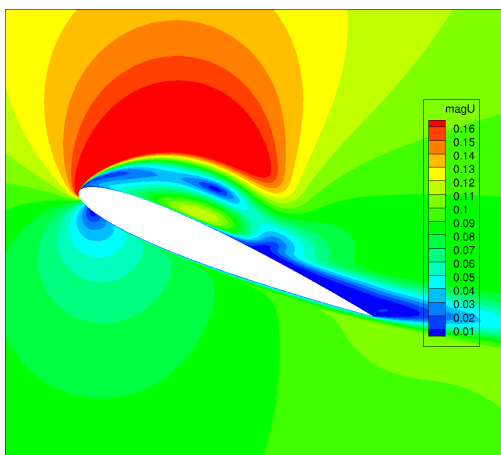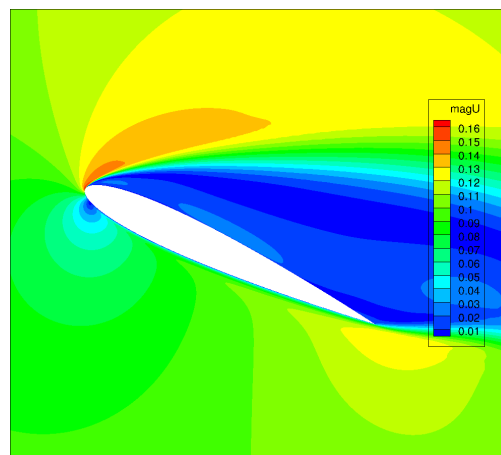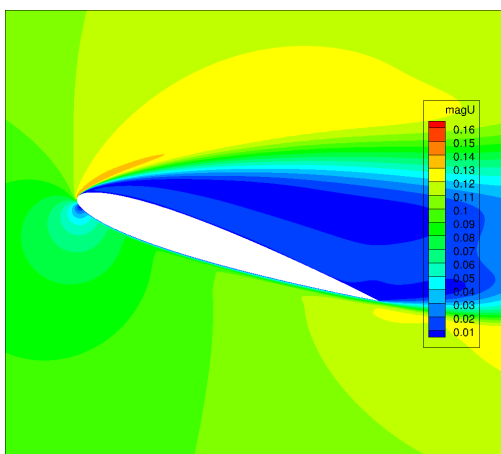
(e) Approximately 20° dec.
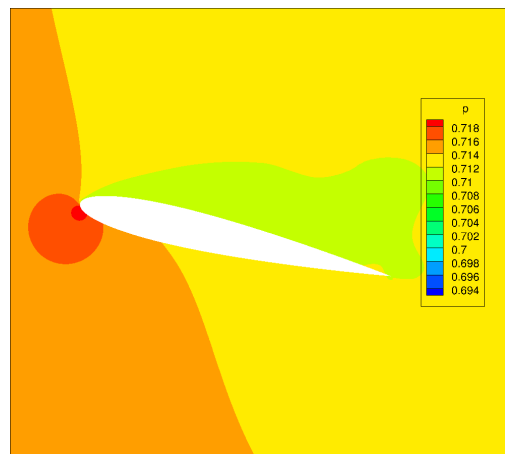
(f) Approximately 15° dec.

Figure 22: Color contour plots at various AoA of non-dimensional pressure for the first half of the cycle of an oscillating airfoil. The wall of the airfoil is represented with the IBM.
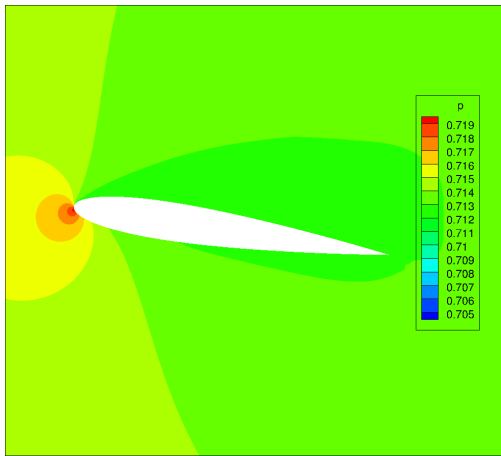
(a) Approximately 10° dec.
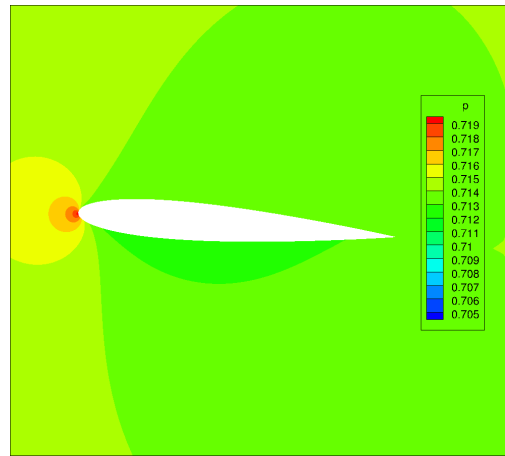
(b) Approximately 5° dec.
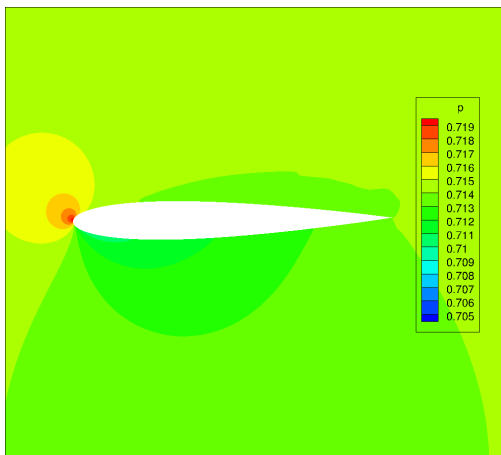
(c) Approximately 0° dec.

(d) Approximately -5° dec.

(e) Approximately 0° inc.

(f) Approximately 5° inc.

Figure 23: Color contour plots of non-dimensional pressure at various AoA for the second half of the cycle of an oscillating airfoil. The wall of the airfoil is represented with the IBM.
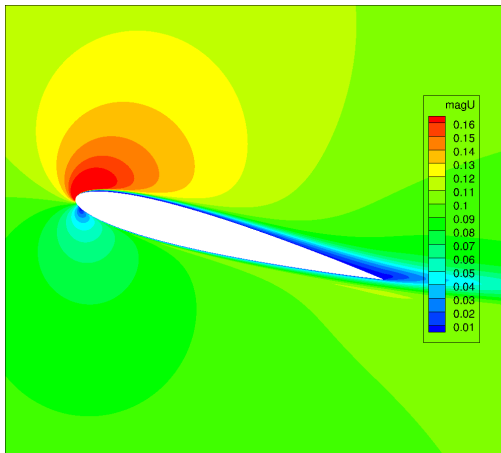
(a) Approximately 10° inc.
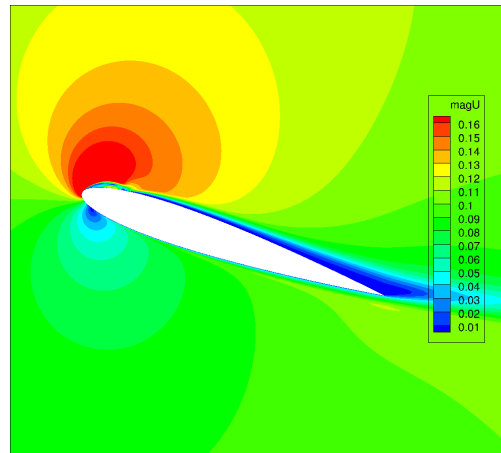
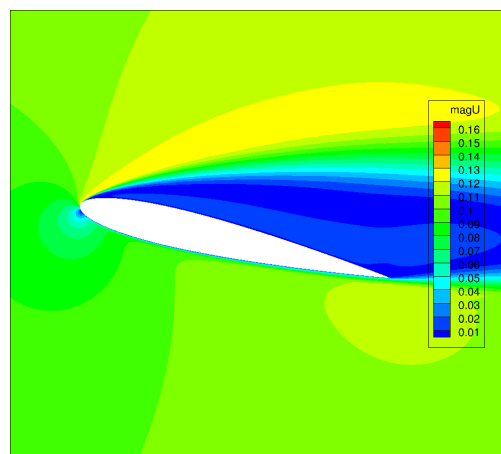(b) Approximately 15° inc.

(c) Approximately 20° inc.
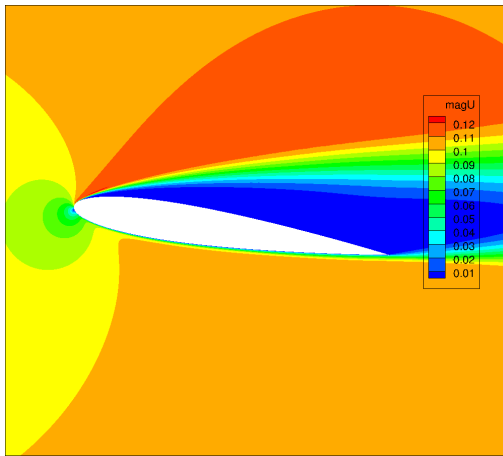
(d) Approximately 25° inc.
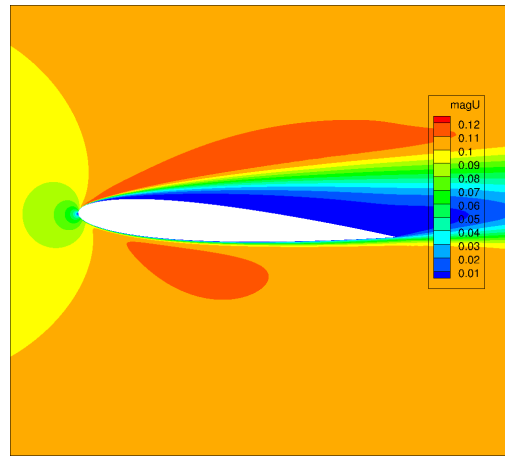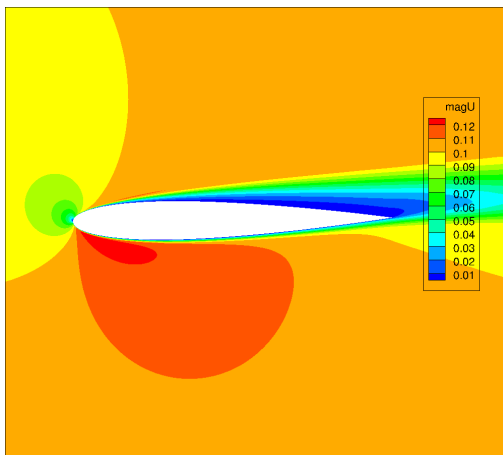
(e) Approximately 20° dec.

(f) Approximately 15° dec.

Figure 24: Color contour plots of magnitude of non-dimensional velocity at various AoA for the first half of the cycle of an oscillating airfoil. The wall of the airfoil is represented with the IBM.
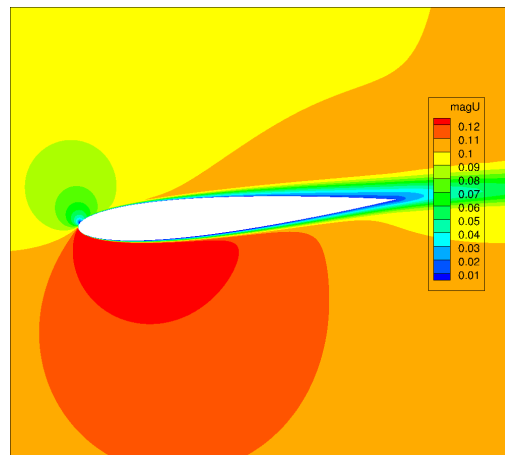
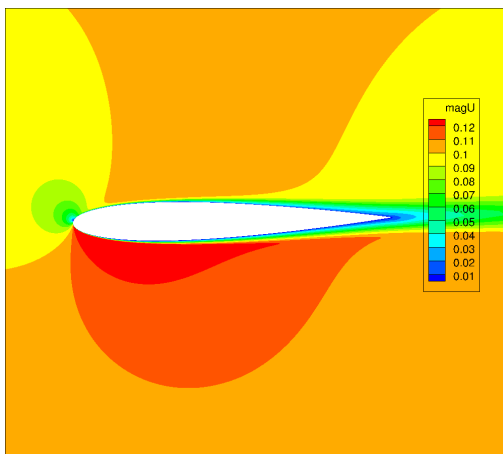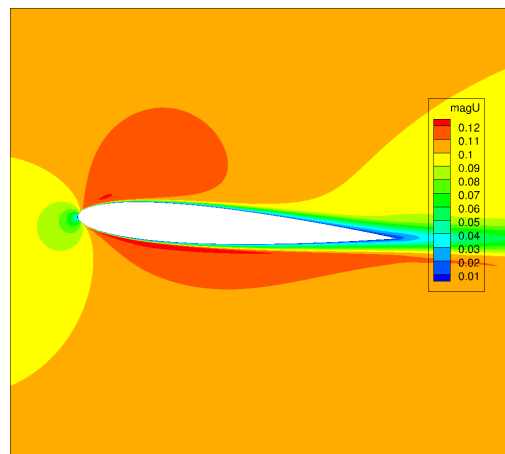(a) Approximately 10° dec.

(b) Approximately 5° dec.

(c) Approximately 0° dec.

(d) Approximately -5° dec.

(e) Approximately 0° inc.

(f) Approximately 5° inc.

Figure 25: Color contour plots of magnitude of non-dimensional velocity pressure at various AoA for the second half of the cycle of an oscillating airfoil (the higher angles of attack). The wall of the airfoil is represented with the IBM.

V.   Conclusions

In this paper an Immersed Boundary Method (IBM) is used to represent the body in computational fluid dynamic simulations with a goal of simulating the flow field around aerodynamic surfaces where ice as accreted. The IBM method, which is a non-cut-cell approach, has been implemented in FUN3D, which is a NASA developed unstructured CFD code. The Immersed Boundary Method removes the need for computational cells to undergo geometric changes as an ice shape accumulates on aerodynamic walls. Instead the IBM alters the governing equations for particular cells making them behave in the solution algorithm as a solid, viscous body. This can allow cells to be taken dynamically from the computational domain and converted to a "solid" and easily accommodate the change in shape resulting from ice accumulation on a surface.

Implementation and development of this IBM into FUN3D has been described. This includes the modifications to the governing equations, node identification by use of a program used for an overset method (SUGGAR++ and DiRTlib), and the modifications to the turbulence model. Preliminary validation of this method has been shown herein for two-dimensional airfoils in various configurations including a clean airfoil, an iced airfoil, and a dynamic airfoil. The results of these simulations are shown to compare satisfactorily with experimental results and results from FUN3D with body conforming grids.

References

[1]Wright, W. B. and Bidwell, C. S., "Additional Improvements to the NASA Lewis ICcretion Code LEWICE," *NASA Technical Memorandum*, Vol. NASA TM-106849, 1995.

[2]Bidwell, C. S. and Potapczuk, M. G., "Users Manual for the NASA Lewis Three-Dimensional Ice Accretion Code (LEWICE3D)," *NASA Technical Memorandum*, No. NASA TM 105974, 1993.

[3]Beaugendre, H., Morency, F., and Habashi, W. G., "FENSAP-ICE's Three-Dimensioanl In-Flight Ice Accretion Module: ICE3D," *Journal of Aircraft*, Vol. 40, No. 2, 2003.

[4]Kinzel, M. P., Sarofeen, C. M., Noack, R. W., and Kreeger, R. E., "A Finite-Volume Approach to Modeling Ice Accretion," *28th AIAA Applied Aerodynamics Conference*, No. 2010-4230, 2010.

[5]Chi, X., Zhu, B., and Shih, T. I.-P., "Computing Aerodynamic Performance of a 2D Iced Airfoil: Blocking Topology and Grid Generation," *40th Aerospace Sciences Meeting and Exhibit*, No. 2002-0381, 2002.

[6]Wright, W., "User's Manual for LEWICE Version 3.2," *CASI*, No. NASA/CR-2008-214255, 2008.

[7]Biedron, R. T. and Thomas, J. L., "Recent Enhancements To The FUN3D Flow Solver For Moving-Mesh Applications," *AIAA*, , No. 1360, 2009.

[8]Anderson, W. K. and Bonhaus, D. L., "An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids," *Computers Fluids*, Vol. 23, No. 1, 1994, pp. 1–21.

[9]Biedron, R. T., Vasta, V. N., and Atkins, H. L., "Simulation of Unsteady Flows Using an Unstructured Navier-Stokes Solver on Moving and Stationary Grids," *23rd AIAA Applied Aerodynamics Conference*, No. 5093, 2005.

[10]Noack, R. W., "SUGGAR: a General Capability for Moving Body Overset Grid Assembly," *17th AIAA CFD Conference*, No. 2005-5117, 2006.

[11]Noack, R. W., "DiRTlib: A Library to Add an Overset Capability to Your Flow Solver," *17th AIAA CFD Conference*, No. 2005-5116, 2006.

[12]Cho, Y., Boluriaan, S., and Morris, P. J., "Immersed Boundary Method for Viscous Flow Around Moving Bodies," *44th AIAA Aerospace Sciences Meeting and Exhibit*, No. AIAA 2006-1089, 2006.

[13]Menter, F. R., "Improved Two-Equation k-ω Turbulence Models for Aerodyanmic Flows," *NASA Technical Memorandum*,No. 103975, 1992.

[14]Kim, J., Kim, D., and Choi, H., "An Immersed-Boundary Finite-Volume Method for Simulations of Flow in Complex Geometries," *Journal of Computational Physics*, No. 0021-9991/01, 2001, pp. 132–150.

[15]Busch, G., Broeren, A., and Bragg, M., "Aerodynamic Simulation of a Horn-ice Accretion on a Subscale Model," .

[16]Lee, T. and Gerontakos, P., "Investigation of Flow Over an Oscillating Airfoil," *Journal of Fluid Mechanics*, Vol. 512, 2004, pp. 313–341.

1. REPORT DATE *(DD-MM-YYYY)* 01-07-2013	2. REPORT TYPE Technical Memorandum	3. DATES COVERED *(From - To)*

4. TITLE AND SUBTITLE	5a. CONTRACT NUMBER
A Non-Cut Cell Immersed Boundary Method for Use in Icing Simulations	NNX08AW92A
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)	5d. PROJECT NUMBER
Sarofeen, Christian, M.; Noack, Ralph, W.; Kreeger, Richard, E.	5e. TASK NUMBER
	5f. WORK UNIT NUMBER WBS 877868.02.07.03.03.01

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	8. PERFORMING ORGANIZATION REPORT NUMBER
National Aeronautics and Space Administration John H. Glenn Research Center at Lewis Field Cleveland, Ohio 44135-3191	E-18692

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	10. SPONSORING/MONITOR'S ACRONYM(S)
National Aeronautics and Space Administration Washington, DC 20546-0001	NASA
	11. SPONSORING/MONITORING REPORT NUMBER NASA/TM-2013-217881

12. DISTRIBUTION/AVAILABILITY STATEMENT
Unclassified-Unlimited Subject Category: 02 Available electronically at http://www.sti.nasa.gov This publication is available from the NASA Center for AeroSpace Information, 443-757-5802

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This paper describes a computational fluid dynamic method used for modelling changes in aircraft geometry due to icing. While an aircraft undergoes icing, the accumulated ice results in a geometric alteration of the aerodynamic surfaces. In computational simulations for icing, it is necessary that the corresponding geometric change is taken into consideration. The method used, herein, for the representation of the geometric change due to icing is a non-cut cell Immersed Boundary Method (IBM). Computational cells that are in a body fitted grid of a clean aerodynamic geometry that are inside a predicted ice formation are identified. An IBM is then used to change these cells from being active computational cells to having properties of viscous solid bodies. This method has been implemented in the NASA developed node centered, finite volume computational fluid dynamics code, FUN3D. The presented capability is tested for two-dimensional airfoils including a clean airfoil, an iced airfoil, and an airfoil in harmonic pitching motion about its quarter chord. For these simulations velocity contours, pressure distributions, coefficients of lift, coefficients of drag, and coefficients of pitching moment about the airfoil's quarter chord are computed and used for comparison against experimental results, a higher order panel method code with viscous effects, XFOIL, and the results from FUN3D's original solution process. The results of the IBM simulations show that the accuracy of the IBM compares satisfactorily with the experimental results, XFOIL results, and the results from FUN3D's original solution process.

15. SUBJECT TERMS
Icing; Scaling; Rotorcraft

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON STI Help Desk (email:help@sti.nasa.gov)
a. REPORT U	b. ABSTRACT U	c. THIS PAGE U	UU	38	19b. TELEPHONE NUMBER *(include area code)* 443-757-5802