



HAL
open science

A Penalization Method for Eulerian Droplet Impingement Simulations towards Icing Applications

Pierre Lavoie, Emmanuel Radenac, Ghislain Blanchard, Eric Laurendeau,
Philippe Villedieu

► **To cite this version:**

Pierre Lavoie, Emmanuel Radenac, Ghislain Blanchard, Eric Laurendeau, Philippe Villedieu. A Penalization Method for Eulerian Droplet Impingement Simulations towards Icing Applications. AIAA Scitech 2021 Forum (the American Institute of Aeronautics and Astronautics), Jan 2021, VIRTUAL EVENT, United States. pp.AIAA 2021-1442, 10.2514/6.2021-1442 . hal-03172019

HAL Id: hal-03172019

<https://hal.science/hal-03172019>

Submitted on 17 Mar 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Penalization Method for Eulerian Droplet Impingement Simulations towards Icing Applications

Pierre Lavoie^{*1,2}, Emmanuel Radenac², Ghislain Blanchard², Eric Laurendeau¹, and Philippe Villedieu²

¹*Department of Mechanical Engineering, Polytechnique Montreal, C.P. 6079, succ. Centre-ville, Montreal (Qc), H3C 3A7, Canada*

²*ONERA / DMPE, Université de Toulouse, F-31055 Toulouse, France*

The numerical prediction of in-flight ice accretion generally involves geometry updates and re-meshing as the ice builds up. However, the generation of body-fitted meshes around complex ice shapes is not trivial and can be repeated several times to obtain the final ice shape. The use of an immersed boundary method can simplify the mesh generation and help in the automation of the ice accretion process. This paper studies the application of an immersed boundary method to Eulerian droplet impingement simulations. A penalization method is suggested requiring only the addition of source terms in the continuous form of the equations. The wall boundary condition must be treated with care to avoid droplets re-injection in the computational domain from a solid boundary. This is solved by the introduction of a droplet mask function in addition to the usual solid mask, providing an automatic detection of the wall boundary condition and therefore avoiding droplet re-injection. The approach is tested on canonical cylinder cases and on more realistic NACA0012 airfoil and ice horn cases. The results show that the solution from a body-fitted simulation can be reproduced using the penalization method.

I. Introduction

Numerical tools for the prediction of in-flight ice accretion have been developed and used for many years (e.g. [1]). Ice accretion is an unsteady multi-physics process where supercooled water droplets impinge on a cold surface [2] (e.g. aircraft wings, tail, etc.) and might freeze upon impact or run back and freeze farther downstream. Typical tools for the prediction of ice accretion segregate the simulation of the physics in different modules (e.g. LEWICE [3]) that are called sequentially in a quasi-steady approach:

- 1) mesh generation;
- 2) computation of the aerodynamics;
- 3) computation of the droplet trajectories and impingement rates;
- 4) evaluation of the convective heat transfer at the wall;
- 5) computation of the ice accretion (mass and heat balance);
- 6) geometry update (the ice shape is generated).

This process (modules 1 to 6) is usually repeated several times to increase the accuracy of the ice accretion prediction using a multi-step approach [4], requiring mesh generation for each step.

When using body-fitted (BF) meshes, the effort is spent on generating a good quality mesh that matches the geometry. This process can be difficult to automate on complex ice geometries, especially in 3D. On the other hand, when using an Immersed Boundary Method (IBM), the mesh generation can be much simpler (e.g. Cartesian grid) as the geometry is allowed to arbitrarily cut through the mesh. In this case the effort is spent on the correct imposition of the boundary condition on the immersed boundary. It is thus envisioned to replace the classical body-fitted meshes by the use of an immersed boundary method which could ease the automation of the ice accretion process.

Although there is a higher benefit in using Immersed Boundary Methods for 3D ice accretion, the developments are first performed in 2D using ONERA's IGLOO2D ice accretion suite [5]. For the aerodynamics, IGLOO2D uses a Euler flow solver combined with an integral boundary layer code. For the evaluation of the droplet trajectories, both Lagrangian and Eulerian solvers are available. An immersed boundary method (penalization) was previously developed for the Euler equations and presented in [6]. As a continuation, the objective of this paper is to apply the penalization method to the Eulerian solver for the droplet trajectories.

*Corresponding author: pierre.lavoie@polymtl.ca

A variety of Immersed Boundary Methods are available from the literature, ranging from the geometric Cut-Cell approach [7–9] to discrete [10–12] and continuous methods [13, 14]. Continuous immersed boundary methods have the advantage of being independent of the discretization and numerical method. Furthermore, continuous approaches such as the penalization method of [14] are appealing for their simplicity of implementation. The penalization method was applied to incompressible Navier-Stokes equations [14, 15] and extended to compressible flows [16–18]. A variant of the penalization method, the Characteristic-Based Volume Penalization, was developed by [19] and applied to the Navier-Stokes and Euler equations [6, 20].

Applications of an immersed boundary method for the Eulerian droplet trajectory solver are few. A discrete method (a type of ghost-cell approach) is used by [21] to perform both 2D and 3D simulations on Cartesian grids. A similar approach is followed by [22] where a discrete approach is used in combination with a level-set. According to the authors' knowledge, there is no application of a penalization method for Eulerian trajectory solvers in the literature.

The application of the penalization method to the Eulerian droplet trajectory equations is not straightforward because of the nature of the wall boundary condition that changes along the wall according to the droplet trajectories. It is quite simple to deal with this situation when using ghost-cells, but it must be treated with care with the penalization method as droplets could otherwise enter the solid body and be re-injected in the field downstream. This paper suggests a way to apply the penalization method to the droplet equations which avoids droplet re-injection by using a double mask function.

The paper is structured in three main sections. First the Eulerian droplet equations and their application for Body-Fitted simulations are reviewed. Second, a penalization approach is suggested for droplet impingement and the double mask function is explained. Third, the verification of the method is made on canonical cylinder cases, on a NACA0012 clean airfoil in icing conditions and on an ice accreted GLC305 airfoil exhibiting an ice horn.

II. Eulerian Droplets Impingement

For ice accretion prediction, the fundamental information to be retrieved is the droplet impingement rate (\dot{m}_{imp}) on the body (e.g. an airfoil). This is generally computed in terms of collection efficiency (β) which can be seen as a non-dimensional impingement rate. The Eulerian approach for the evaluation of the droplet impingement consists in retrieving the volume fraction of water (α) and the droplet velocity (\mathbf{v}_d) in the field surrounding the body of interest (e.g. an airfoil). A system of PDE is then solved for the droplets assuming a one-way coupling with the aerodynamic field.

A. Governing Equations

The non-dimensional continuity and momentum equations for the droplets are respectively in non-conservative form [23]:

$$\begin{aligned} \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}_d) &= 0 \\ \frac{\partial \mathbf{v}_d}{\partial t} + \mathbf{v}_d \cdot \nabla \mathbf{v}_d &= \frac{C_D Re_d}{24 Stk} (\mathbf{v}_a - \mathbf{v}_d) + \left(1 - \frac{\rho_a}{\rho_d}\right) \frac{1}{Fr^2} \mathbf{g} \end{aligned} \quad (1)$$

In conservative form the equations become:

$$\begin{aligned} \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}_d) &= 0 \\ \frac{\partial (\alpha \mathbf{v}_d)}{\partial t} + \nabla \cdot (\alpha \mathbf{v}_d \otimes \mathbf{v}_d) &= \frac{C_D Re_d}{24 Stk} \alpha (\mathbf{v}_a - \mathbf{v}_d) + \alpha \left(1 - \frac{\rho_a}{\rho_d}\right) \frac{1}{Fr^2} \mathbf{g} \end{aligned} \quad (2)$$

Where α is the non-dimensional volume fraction of water, \mathbf{v}_d is the non-dimensional droplets velocity, \mathbf{v}_a is the non-dimensional air velocity, ρ_d is the droplets density (density of water), ρ_a is the air density and C_D is the droplets drag coefficient. The droplets Reynolds number (Re_d), the Stokes number (Stk) and the Froude number (Fr) are

respectively defined as:

$$Re_d = \frac{\rho_a \|\mathbf{v}_a - \mathbf{v}_d\| D_d}{\mu} \quad (3)$$

$$Stk = \frac{\rho_d D_d^2 U_\infty}{18L\mu} \quad (4)$$

$$Fr = \frac{U_\infty}{\sqrt{Lg}} \quad (5)$$

where D_d is the droplet diameter, μ the dynamic viscosity of air and L a characteristic dimension (e.g. the chord length for an airfoil). The drag model of Schiller and Naumann [24] is used for the droplets which are assumed to remain spherical:

$$C_D = \begin{cases} \frac{24}{Re_d} (1 + 0.15 Re_d^{0.687}) & Re_d \leq 1000 \\ 0.4 & Re_d > 1000 \end{cases} \quad (6)$$

The Eulerian formulation of the equations allows simple evaluation of the collection efficiency (β) at the wall as α and \mathbf{v}_d are known everywhere in the field and \mathbf{n}_{bc} is known from the geometry.

$$\beta = \alpha \mathbf{v}_d \cdot \mathbf{n}_{bc} \quad (7)$$

The impinging water flux is then retrieved by:

$$\dot{m}_{imp} = (LWC) U_\infty \beta \quad (8)$$

where LWC is the Liquid Water Content and U_∞ the magnitude of the freestream velocity.

The complete equations are presented here for generality. However for the remainder of this paper the gravity term is neglected and thus only the drag force will act on the droplets.

B. Boundary Conditions

One important aspect of the Eulerian model for droplet impingement is the treatment of the boundary conditions. The system of equations is hyperbolic and therefore boundary conditions are only required for incoming characteristics [23, 25] (Fig. 1). Additional numerical boundary schemes compatible with the physics of the problem are applied where boundary conditions are not imposed. For instance, a zero-order extrapolation is used for outgoing characteristics. Assuming the boundary normals are pointing out of the computational domain (Fig. 1), the boundary condition for the droplet equations can be formulated as follows.

Far-field :

$$\left. \begin{array}{l} \alpha = \alpha_\infty \\ \mathbf{v}_d = \mathbf{v}_\infty \end{array} \right\} \text{if } \mathbf{v}_d \cdot \mathbf{n}_{bc} \leq 0 \quad (9)$$

Wall :

$$\left. \begin{array}{l} \alpha = 0 \\ \mathbf{v}_d = 0 \end{array} \right\} \text{if } \mathbf{v}_d \cdot \mathbf{n}_{bc} \leq 0 \quad (10)$$

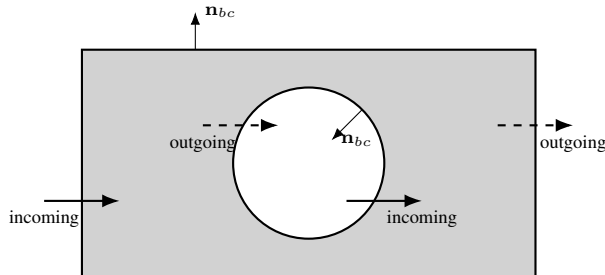


Fig. 1 Boundary Condition Diagram for Droplets Incoming and Outgoing Information

In this paper, the focus is on the wall boundary condition as only this condition is treated with the penalization method. When the droplets enter the computational domain from the solid ($\mathbf{v}_d \cdot \mathbf{n}_{bc} \leq 0$), a boundary condition is applied on the primitive variables (Eq. (10)), enforcing a null flux. Otherwise when droplets impinge the body ($\mathbf{v}_d \cdot \mathbf{n}_{bc} > 0$), an extension of the primitive variable is performed. For body-fitted meshes, this is typically done by setting $\mathbf{n}_{bc} \cdot \nabla \alpha = 0$ and $\mathbf{n}_{bc} \cdot \nabla \mathbf{v}_d = 0$ at the boundary, but a slightly different approach is used for the penalization method in the next section.

III. Penalization Method

This section discusses the definition of the immersed boundary and presents the penalization method applied to the droplet equations.

A. Immersed Boundary Representation

For this paper, the immersed boundary is defined by a discrete list of nodes (2D) and its location in the mesh is defined by the level-set $\phi = 0$, where ϕ is the signed distance field from the immersed boundary. Values of ϕ are computed using a geometric approach : evaluating the minimum projected distance to the edges forming the immersed boundary [26].

The sign of ϕ is determined by a ray casting algorithm [26] where the immersed boundary is considered as a closed body. Using a signed distance field leads to a simple evaluation of the normals (\mathbf{n}_ϕ) and curvature (κ) of the immersed boundary using:

$$\mathbf{n}_\phi = -\frac{\nabla \phi}{\|\nabla \phi\|} \quad (11)$$

$$\kappa = \nabla \cdot \mathbf{n}_\phi. \quad (12)$$

Note that the normal based on ϕ has a negative sign in order to point towards the solid zone ($\phi < 0$). This is useful in the definition of the penalization method presented next.

To extract the data at the immersed boundary (e.g. α , \mathbf{v}_d), a weighted least square interpolation at the discrete nodes defining the immersed boundary is used. The interpolation stencil is determined firstly by identifying the cell containing the immersed boundary node, and secondly by storing the extended neighborhood of this cell (neighbors sharing a node with the IB cell). An inverse distance weighting Eq. (13) is used for the interpolation with a smoothing parameter ($\epsilon = 0.5\Delta x_J$) to avoid dividing by a small value when cell centers are near the interpolation point. The weight for the interpolation point P and a cell-center J (part of the interpolation stencil) is written as:

$$w_J = \frac{1}{\sqrt{\|\mathbf{r}_{PJ}\|^2 + \epsilon^2}} \quad (13)$$

where Δx_J is the characteristic size of cell J and $\|\mathbf{r}_{PJ}\|$ is the distance between P and J .

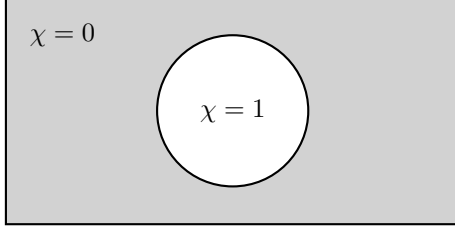
In this paper, the collection efficiency (β) is evaluated at the immersed boundary (IB) in two steps. First, the primitive variables (α , \mathbf{v}_d) and the normals (\mathbf{n}_ϕ) are interpolated at the IB discrete nodes using the weighted least square method. Second, the collection efficiency is evaluated with $\beta = \alpha \mathbf{v}_d \cdot \mathbf{n}_\phi$. As the interpolation stencil includes solid cells, the penalization method described in the following sections must be designed so the interpolation stencil in the vicinity of the solid/fluid interface is filled with valid data to perform the interpolation (e.g. valid data in the solid).

B. Application to the Droplet Equations

The Volume Penalization method [14] consists in adding source terms in the continuous form of the equation to enforce the desired boundary condition. The source terms are activated/deactivated using a mask function (χ) equal to unity in the solid and zero in the fluid. In this way, only the solid is penalized and the usual equations are retrieved in the fluid. A penalization parameter (η) ensures the boundary condition is enforced accurately. The volume penalization enforces the boundary conditions at the cell centers surrounding the immersed boundary (for a finite volume cell-centered method). Thus only an approximation of the IB is seen by the solver and this typically limits the penalization method to 1st order accuracy in space. However, second order accurate approaches can be implemented by a discretization of the penalization source term [27, 28]. In this paper, the former approach is used.

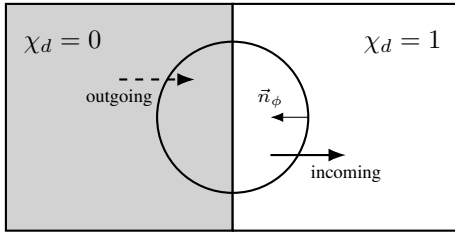
For the droplet equations, no boundary condition should be applied for impinging droplets and a Dirichlet condition must be enforced in the shadow zone ($\mathbf{v}_d \cdot \mathbf{n}_\phi \leq 0$) to avoid droplet re-injection in the computational domain (see

Eq. (10) and Fig. 1). A typical penalization method uses a fluid/solid mask function (χ) to apply the penalization term everywhere in the solid. In this paper a sharp Heaviside function (Fig. 2 and Eq. (14)) is used. However this is not sufficient as the Dirichlet condition would then also be applied in the impingement zone ($\mathbf{v}_d \cdot \mathbf{n}_\phi > 0$), where no boundary condition is required. To solve this issue, an inflow/outflow droplet mask function (χ_d) is also defined as a sharp Heaviside function (Fig. 3 and Eq. (15)).



$$\chi = \begin{cases} 0 & \phi \geq 0 \\ 1 & \phi < 0 \end{cases} \quad (14)$$

Fig. 2 mask function (Solid: $\chi = 1$, Fluid: $\chi = 0$)



$$\chi_d = \begin{cases} 0 & \alpha \mathbf{v}_d \cdot \mathbf{n}_\phi \geq 0 \\ 1 & \alpha \mathbf{v}_d \cdot \mathbf{n}_\phi < 0 \end{cases} \quad (15)$$

Fig. 3 droplet mask function (shadow zone: $\chi_d = 1$, impingement zone: $\chi_d = 0$)

To apply the penalization term correctly, the two mask functions are combined ($\chi\chi_d$) which allows penalizing the equations only in the solid shadow zones (Fig. 4).

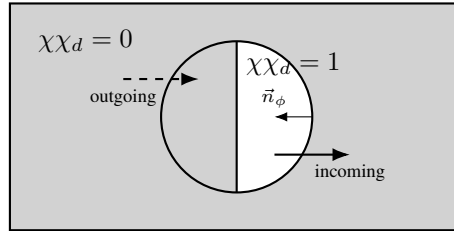


Fig. 4 combined droplet mask function (shadow solid: $\chi\chi_d = 1$, impingement solid/fluid: $\chi\chi_d = 0$)

With the combined mask function defined, the penalized droplet equations can be obtained. The penalization terms are first applied to the non-conservative form of the equations so the boundary conditions are applied on the primitive variables. Then, the equations are transformed in conservative form for implementation. The non-conservative form can be expressed as, including the penalization terms:

$$\begin{aligned} \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}_d) &= -\frac{\chi\chi_d}{\eta} \alpha \\ \frac{\partial \mathbf{v}_d}{\partial t} + \mathbf{v}_d \cdot \nabla \mathbf{v}_d &= \xi \left(\frac{C_D Re_d}{24 Stk} (\mathbf{v}_a - \mathbf{v}_d) + \left(1 - \frac{\rho_a}{\rho_d} \right) \frac{1}{Fr^2} \mathbf{g} \right) - \frac{\chi\chi_d}{\eta} \mathbf{v}_d \end{aligned} \quad (16)$$

Implementation-wise the conservative form of the equations is used and it can be written as:

$$\begin{aligned} \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{v}_d) &= -\frac{\chi \chi_d}{\eta} \alpha \\ \frac{\partial (\alpha \mathbf{v}_d)}{\partial t} + \nabla \cdot (\alpha \mathbf{v}_d \otimes \mathbf{v}_d) &= \xi \left(\frac{C_D Re_d}{24 Stk} \alpha (\mathbf{v}_a - \mathbf{v}_d) + \alpha \left(1 - \frac{\rho_a}{\rho_d} \right) \frac{1}{Fr^2} \mathbf{g} \right) - 2 \frac{\chi \chi_d}{\eta} \alpha \mathbf{v}_d \end{aligned} \quad (17)$$

For droplets traveling from the solid to the fluid zone ($\chi = \chi_d = 1$), the penalization terms drive the volume fraction of water (α) and the droplet velocity (\mathbf{v}_d) to zero. Otherwise when either χ or χ_d are null, no penalization is applied and the physical equations are solved in the solid acting as a natural extension of the droplet characteristics from the fluid to the solid.

When solving Eq. (16) or Eq. (17) in the solid shadow zone ($\chi = \chi_d = 1$), η is selected small enough for the physical droplet equations to be negligible in front of the penalization terms ($\eta \ll 1$). Using the droplet continuity equation as an example, it comes back to solving an ordinary differential equation of the form:

$$\frac{d\alpha}{dt} = -\frac{\alpha}{\eta} \quad (18)$$

The solution of this ODE is a rapidly decaying exponential, meaning $\alpha = 0$ is imposed almost instantaneously:

$$\alpha = A_0 e^{-t/\eta} \quad (19)$$

The penalization parameter η can be seen as a characteristic timescale which must be small ($\eta \ll 1$) in order to enforce the boundary condition. In this paper, it is selected as $\eta = 10^{-8}$.

In this paper, two penalization methods are presented and are both available from Eq. (16) or Eq. (17) using the optional parameter ξ . The Volume Penalization (VP) method is obtained with $\xi = 1$ and the VP-SSO (VP Solid Source Off) method with $\xi = 0$. Option $\xi = 1$ seems to be a natural expression of the penalization method. But it requires setting a value for the gas velocity field in the solid area. No investigation will be made on the ideal value to be applied to this fictitious gas velocity field. If $\xi = 1$, it will be assumed that such a velocity field is available, for example the one provided by an IBM solution on the aerodynamic field. Otherwise, option $\xi = 0$ is available, where the physical source terms are deactivated in the solid zone rendering the droplet solution independent of the gas velocity field.

C. Numerical Method

In this paper the droplet equations are discretized using a cell-centered Finite Volume Method. The fluxes are evaluated with a HLL scheme [29] inspired by the work of [30]. A piecewise linear reconstruction is used to obtain a 2^{nd} order accurate scheme. The flux at the face ($\mathbf{F}_{n,f}$) is obtained from the left (L) and right (R) states using:

$$S_L = \min(\mathbf{v}_{d,L} \cdot \mathbf{n}, \mathbf{v}_{d,R} \cdot \mathbf{n}) \quad (20)$$

$$S_R = \max(\mathbf{v}_{d,L} \cdot \mathbf{n}, \mathbf{v}_{d,R} \cdot \mathbf{n}) \quad (21)$$

$$\mathbf{F}_{n,f} = \begin{cases} \mathbf{F}_L & S_L \geq 0 \\ \mathbf{F}_{HLL} & S_L \leq 0 \leq S_R \\ \mathbf{F}_R & S_R \leq 0 \end{cases} \quad (22)$$

$$\mathbf{F}_{HLL} = \frac{S_R \mathbf{F}_L - S_L \mathbf{F}_R + S_L S_R (\mathbf{W}_R - \mathbf{W}_L)}{S_R - S_L} \quad (23)$$

with

$$\mathbf{W} = \begin{bmatrix} \alpha \\ \alpha u_d \\ \alpha v_d \end{bmatrix}, \quad \mathbf{F} = (\mathbf{v}_d \cdot \mathbf{n}) \mathbf{W} \quad (24)$$

An explicit time scheme with implicit source terms is used to avoid stability issues related to the drag and penalization terms. The gravity term is deactivated for all simulations.

IV. Results

In this section, four test cases on three different configurations are used to assess the penalization method. The simulation parameters are summarized in Table 1.

Table 1 Simulation Parameters

	Cylinder	Airfoil	Ice Horn
Geometry	Cylinder	NACA0012	GLC305
Chord	D=2.0	0.5334	0.9144
LE radius	1.0	8.46e-3	–
AoA	0.0	4.0	4.0
Mach	0.1	0.185	0.273
P_{static}	100kPa	95.61kPa	101.325 kPa
T_{static}	300.0K	245.2K	268.3K
LWC	1.0 g/m ³	1.3 g/m ³	0.54g/m ³
MVD	304.5 μm	20 μm	20μm
Stk	9.688	9.689	–

The parameters for the cylinder are selected to achieve a nearly incompressible flow and maintain a similar Stokes number than encountered in icing simulations. The Stokes number is evaluated against the leading edge radius which can be computed as $R_{LE} = 1.1019(t_c)^2 c$ for a NACA 4-digit airfoil [31], where t_c is the thickness to chord ratio.

In the following sections, when comparing body-fitted and immersed boundary results, an equivalent mesh size is used at the wall to provide a fair comparison. Both VP and VP-SSO approaches will be used. Although no definite rule is given for the definition of the fictitious aerodynamic velocity field, simple choices will be made for this term in order to compare the VP and VP-SSO methods, and to evaluate what maintaining a source term in the immersed area tends to produce. For the airfoil cases, the aerodynamic velocity field is provided by the Euler IBM [6]. For the cylinder cases, the field provided by the potential solution is extended inside the solid domain.

A. No Drag nor Gravity

An interesting test case for the droplet equations occurs when the droplet drag and gravity are set to zero. In such a case, the droplet equations are independent of the aerodynamic field and an analytical solution is retrieved for the collection efficiency (β) as long as an analytical equation is available for the wall geometry.

1. Analytical Solution

Considering a cylinder of radius R_c centered at $(x, y) = (0, 0)$, the analytical solution at the wall is expressed as:

$$\begin{cases} \alpha = \alpha_\infty, & \mathbf{v}_d = \mathbf{v}_\infty & \text{if } \mathbf{v}_d \cdot \mathbf{n}_\phi > 0 \\ \alpha = 0, & \mathbf{v}_d = 0 & \text{if } \mathbf{v}_d \cdot \mathbf{n}_\phi \leq 0 \end{cases} \quad (25)$$

where the droplet velocity is:

$$\mathbf{v}_\infty = U_\infty (\cos(AoA), \sin(AoA)) \quad (26)$$

and for a cylinder the normal to the wall is:

$$\phi = \sqrt{x^2 + y^2} - R_c \quad (27)$$

$$\mathbf{n}_\phi = -\frac{\nabla \phi}{\|\nabla \phi\|} = -\frac{(x, y)}{\sqrt{x^2 + y^2}}. \quad (28)$$

In turn, the collection efficiency β is evaluated as:

$$\beta = \alpha \mathbf{v}_d \cdot \mathbf{n}_\phi \quad (29)$$

It means that for $AoA = 0$, the collection efficiency is defined at the wall as:

$$\beta = \alpha_{\infty} U_{\infty} \max\left(-\frac{x}{R_c}, 0\right) \quad (30)$$

or in polar coordinates:

$$\beta = \alpha_{\infty} U_{\infty} \max(-\cos\theta, 0) \quad (31)$$

For this specific case, the curvilinear distance is $s = (\theta - \pi)R_c$, which is useful to present the results.

2. Meshes

The simulations are performed for a unit radius cylinder using a body-fitted structured mesh and a Cartesian grid for the immersed boundary method. The structured mesh has a wall cell size of 0.04 radius for a total of 16 000 cells with a far-field located at 50 radii from the cylinder. The Cartesian grid has a wall cell size of 0.05 radius for a total of 64 000 cells with a far-field located at 100 radii from the cylinder. Both meshes are illustrated in Fig. 5.

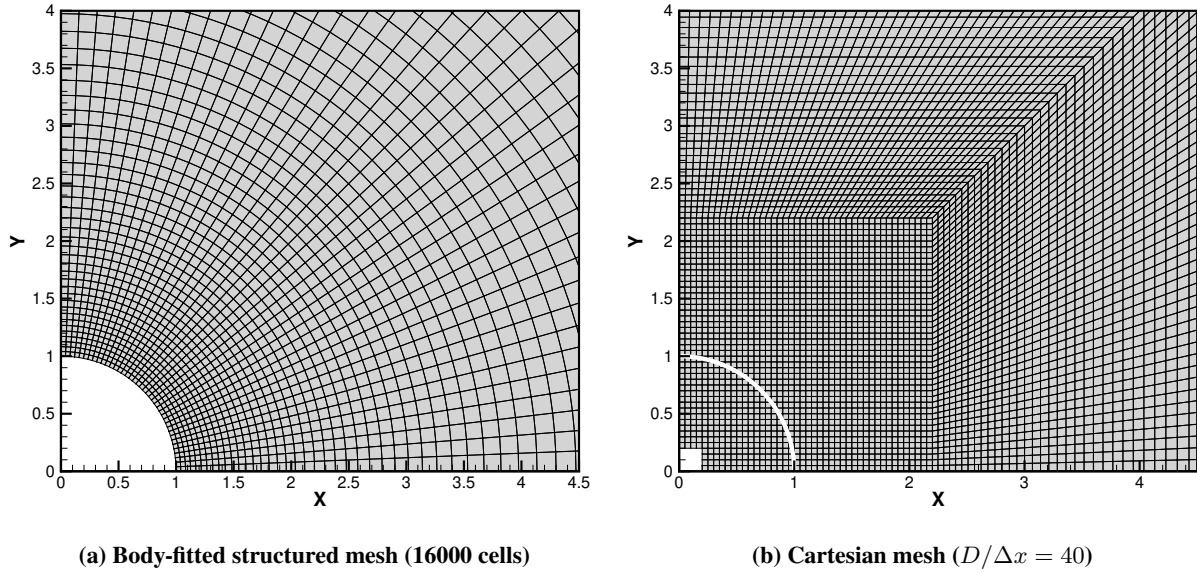


Fig. 5 Meshes for the unit radius cylinder

3. Numerical Solution

The wall collection efficiency (β) is presented in Fig. 6 where the analytical solution is compared with the body-fitted simulation and the penalization method. It shows that the penalization method is able to reproduce the analytical solution as all the curves are overlaid.

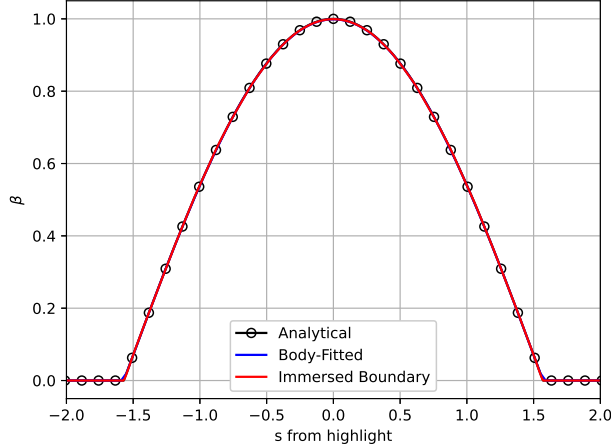


Fig. 6 Analytical wall collection efficiency (β) compared to the Body-Fitted and Immersed boundary results

Note that for this test case the VP and VP-SSO methods are equivalent as the drag and gravity term are deactivated everywhere in the fluid and the solid zones. Thus only one curve labeled "immersed boundary" is illustrated.

The field values for the volume fraction of water (α) are illustrated in Fig. 7 where the impingement ($\alpha > 0$) and shadow zones ($\alpha = 0$) are clearly visible. The immersed cylinder is represented by the white circle in Fig. 7b. This figure shows that the combined mask function behaves as expected for this test case. The upstream solid portion of the cylinder is filled with droplets ($\alpha = \alpha_\infty$, impingement zone) while the downwind solid portion is empty ($\alpha = 0$, shadow zone). Some streamtraces seem to extend in the solid portion of the shadow zone but in this area $\alpha \approx 0$ and $\mathbf{v}_d \approx 0$. The streamtraces are displayed but carry no droplet mass and therefore no droplet is in fact re-injected in the computational domain.

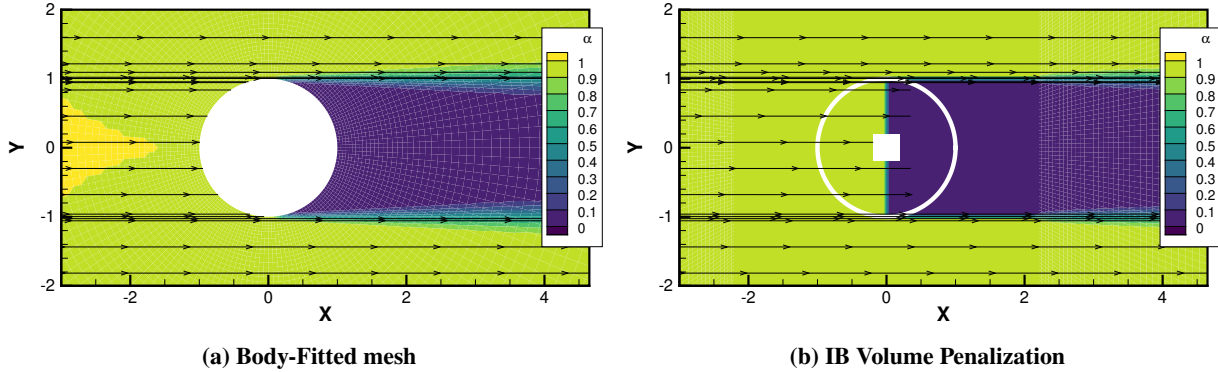


Fig. 7 Volume fraction of water (α) field and droplet streamtraces for the cylinder with no drag nor gravity

This canonical test shows that the penalization method behaves properly. However, as the droplet field is a constant in the impingement zone, this test case is not representative of a real situation and cannot evaluate the quality of the data extraction at the immersed boundary. For a more realistic test, the drag term is reactivated in the next section.

Note that the white square in Fig. 7b is required by the ice accretion suite as it expects a closed body (a closed wall boundary). However, it does not affect the quality of the solution in the field or in the solid. It can also be used to reduce the number of cells inside the solid and therefore the computation cost. In newer versions of IGLOO2D this dummy body is not mandatory.

B. Low Mach flow around a Cylinder

In this section, a mesh convergence study is performed on a cylinder of unit radius using a family of structured meshes for the BF solution and Cartesian grids for the IB solution. Some mesh characteristics are listed in Table 2

where five levels of refinement are used for the structured meshes and four for the Cartesian grids. The finest structured mesh (Level 5) is used to generate a reference solution.

Table 2 Family of meshes

Refinement	Structured			Cartesian		
	Wall Δx [m]	N cell	$D/\Delta x$	Wall Δx [m]	N cell	$D/\Delta x$
Level 0	1.60e-1	1K	12.5	2.00e-1	4K	10
Level 1	8.00e-2	4K	25.0	1.00e-1	16K	20
Level 2	4.00e-2	16K	50.0	5.00e-2	64K	40
Level 3	2.00e-2	64K	100.0	2.50e-2	256K	80
Level 4	1.00e-2	256K	200.0	1.25e-2	1024K	160
Level 5	5.00e-3	1024K	400.0	–	–	–

To avoid possible inaccuracies from the penalized aerodynamic field, the analytical potential flow solution around the cylinder is used. Its definition can be found in the appendix. In this way, the order of convergence for the droplets can be determined independently of the flow solution.

The simulation parameters are summarized in Table 1. In order to avoid that the droplets behave like tracers ($Stk < 0.1$), the MVD is increased to $304.5\mu m$ (compared to $20\mu m$ for the NACA0012). This gives a Stokes number ($Stk \approx 9.7$) close to typical icing conditions, as computed for the NACA0012 case.

The field values for the volume fraction of water (α) are illustrated in Fig. 8 where the VP and VP-SSO methods are compared to the body-fitted simulation. The finest meshes are used for comparison which means refinement level 5 for the structured mesh (BF reference solution) and level 4 for the Cartesian grid (IB). Both VP and VP-SSO methods seem to reproduce the BF solution in the fluid. However, the solution in the solid behaves differently with the VP method, exhibiting a smoother transition from the fluid to the solid zone compared to the VP-SSO. The data is interpolated at the IB with a stencil including both fluid and solid cells. Thus, although the solution on the fluid side is very similar, the differences on the solid side influence the data extraction at the immersed boundary.

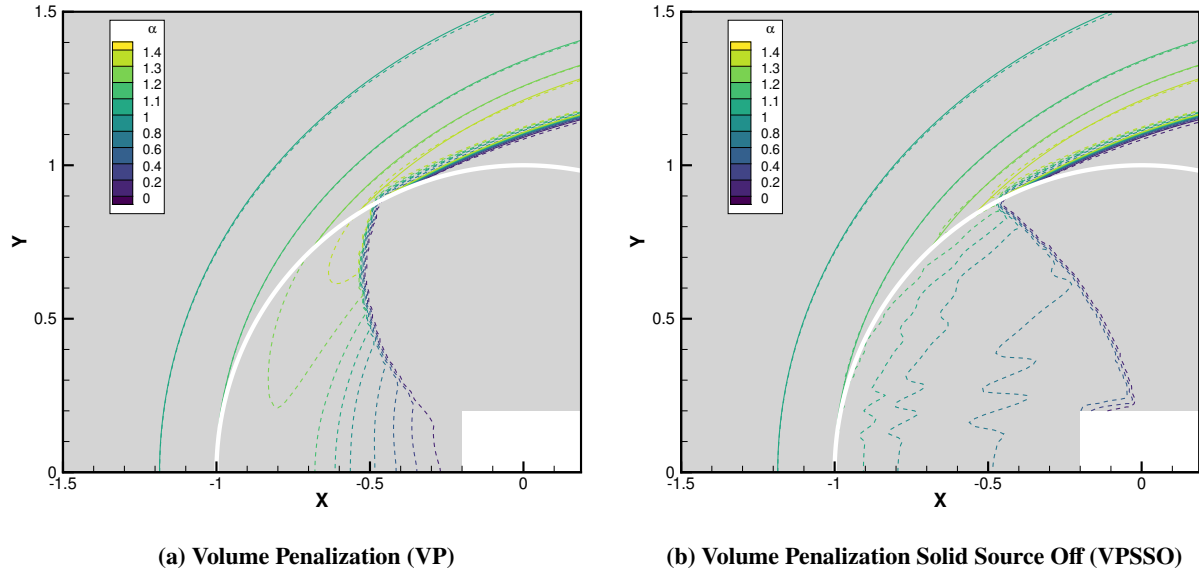


Fig. 8 α field for the BF and IB methods. solid lines: BF, dashed lines: IB

To assess the quality of the data extraction, β is evaluated at the wall for all mesh refinement levels for the BF, VP and VP-SSO methods. On Fig. 9, a mesh converged β distribution is observed from refinement level 4 for the BF method. It justifies the use of refinement level 5 as the reference mesh when comparing with the IBMs.

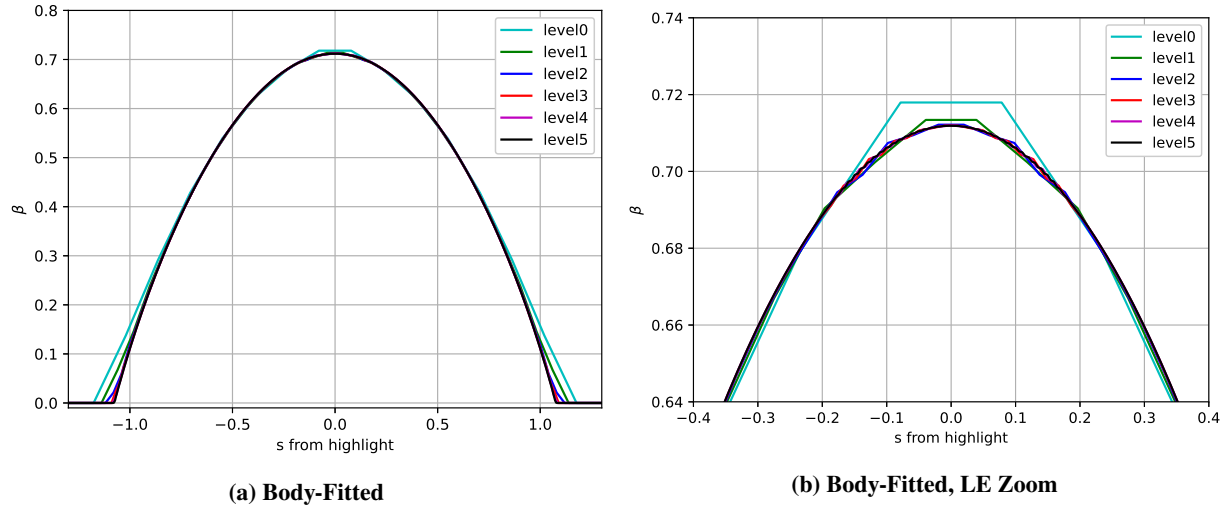


Fig. 9 Collection Efficiency (β) with mesh refinement for the BF mesh.

Fig. 10 shows the collection efficiency with mesh refinement for the VP method. The mesh convergence seems to be slower than for the BF simulation, especially in terms of the maximum β achieved. However, at mesh refinement level 4, the VP solution offers a good match with the reference solution and is also free of oscillations.

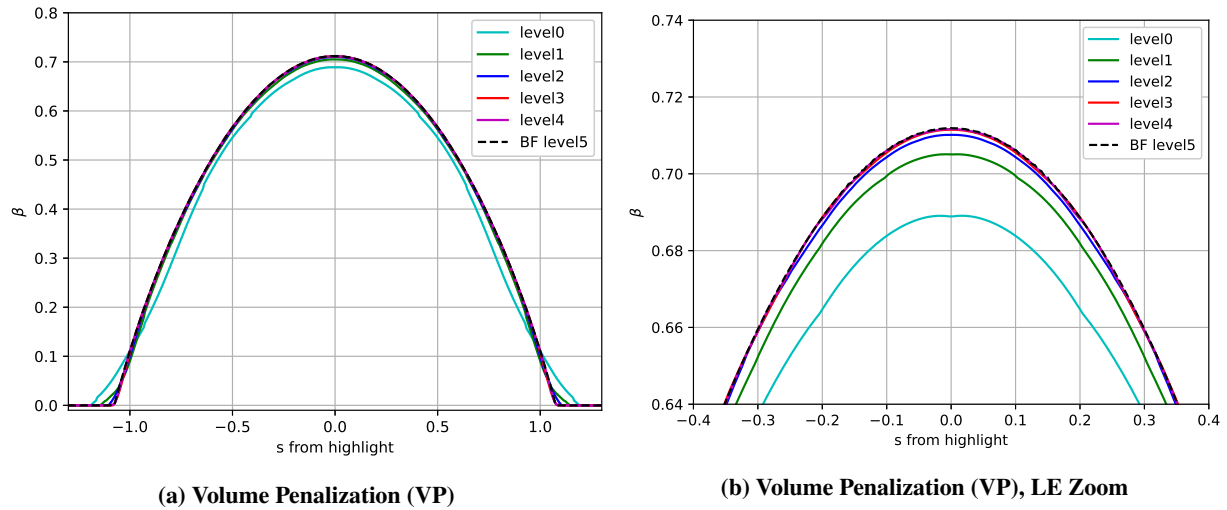


Fig. 10 Collection Efficiency (β) with mesh refinement for the VP method.

For the VP-SSO method (Fig. 11), the β distribution also converges towards the BF reference solution. Contrary to the VP method, the VP-SSO solution exhibits oscillations. The amplitude of these oscillations reduces with mesh refinement but they are still present for the finest mesh. Furthermore, the maximum β does not match the reference solution.

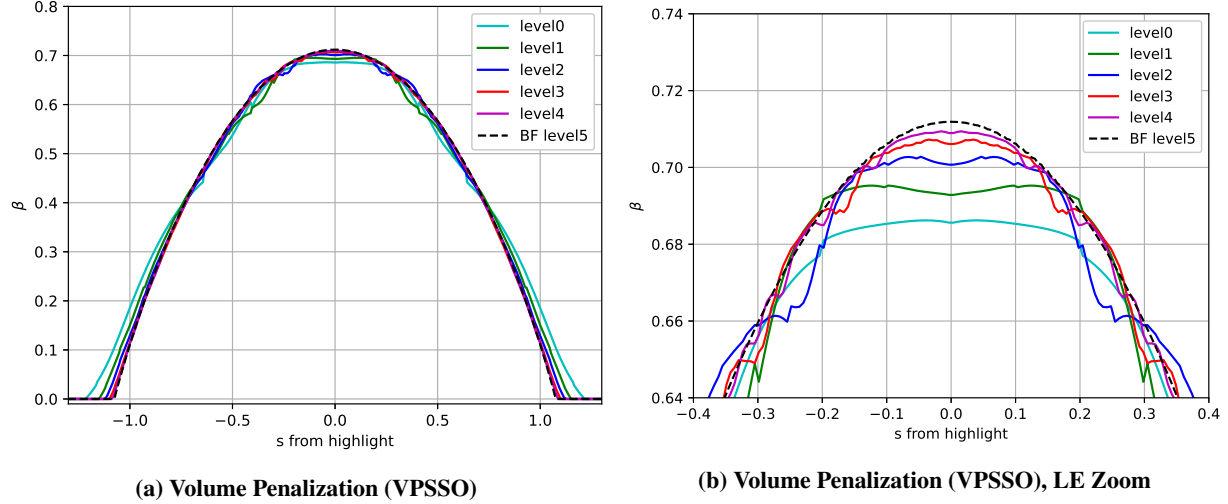


Fig. 11 Collection Efficiency (β) with mesh refinement for the VPSSO method.

The idea behind the VP-SSO method is to avoid having to define an aerodynamic solution in the solid zone, which has no physical meaning and is only required for numerical purposes. This is achieved by deactivating the drag term in the solid. However, the physical flux is still computed in the solid which provides an extension of the droplet solution from the fluid to the solid. As illustrated in Fig. 11b, this method fails to generate a smooth and accurate solution for β . This might be explained by the drag term being turned on and off near the immersed boundary in a staircase manner. On the other hand, the solution is much better when computing the drag term in the solid (VP method), for this particular test case. This seems logical as the drag then influences the droplet field more gradually (no staircase effect). Also, the activation of the physical source term allows a better continuity of the solution near the immersed boundary as previously shown in Fig. 8).

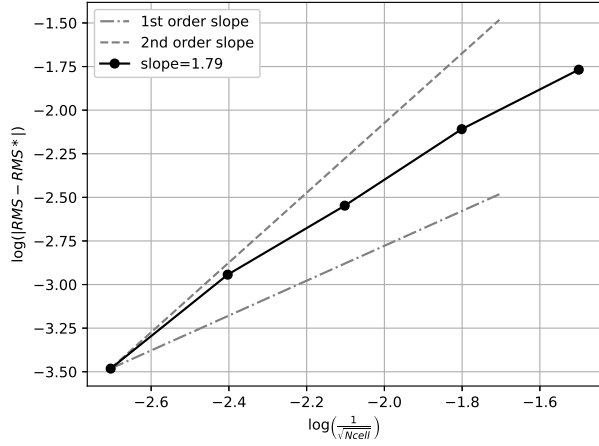
The order of convergence is verified using local (Eq. (32)) and global (Eq. (33)) criteria. The local criterion compares the local collection efficiency (β_i) against the reference solution (body-fitted mesh, refinement level 5) and combines it using a root mean square (RMS). The global criterion is computed as the collection efficiency integral along the cylinder's surface (denoted β_{tot}).

$$\text{RMS}_{ref} = \sqrt{\frac{1}{N} \sum_i^N (\beta_i - \beta_{i,ref})^2} \quad (32)$$

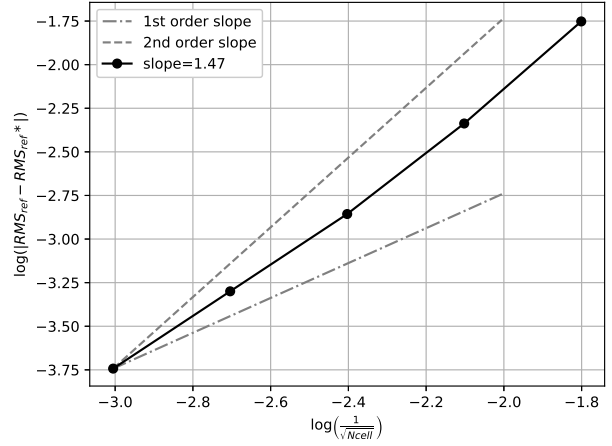
$$\beta_{tot} = \int_0^s (\beta_i) ds \quad (33)$$

The order of convergence p is evaluated using the method described by [32] based on a criterion \mathcal{F} which in this paper is either RMS_{ref} or β_{tot} . This requires a monotonic sequence of \mathcal{F} on a minimum of three meshes (coarse, medium, fine). Then the order p can be solved for using a Newton method and the limiting value of \mathcal{F} as $\Delta x \rightarrow 0$ (denoted \mathcal{F}^*) can be estimated.

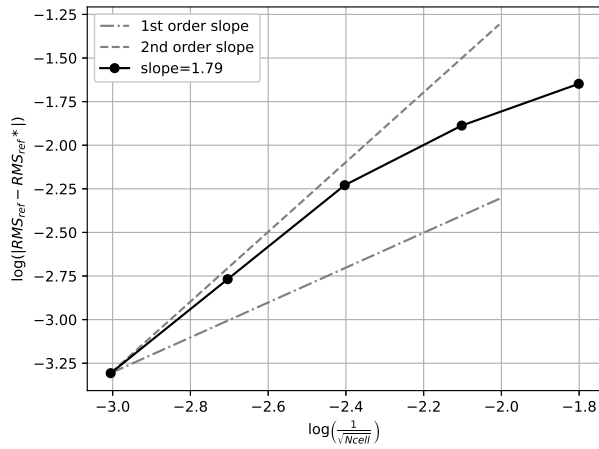
The order of convergence is illustrated on Figs. 12–13 along with the 1st and 2nd order theoretical slopes. Using the structured family of meshes (body-fitted), the order of convergence is $p > 1.7$ for both criteria which is close to 2nd order accuracy. This gives a baseline on what to expect for the penalization methods. The VP method falls short with $p = 1.47$ when using RMS_{ref} but compares well with BF when using β_{tot} ($p = 1.74$). The VP-SSO method compares well with BF for RMS_{ref} ($p = 1.79$). The order of convergence is not shown for the VP-SSO method on β_{tot} since the sequence is not monotonic as the mesh is refined. Thus, p cannot be computed with the current method.



(a) Body-Fitted

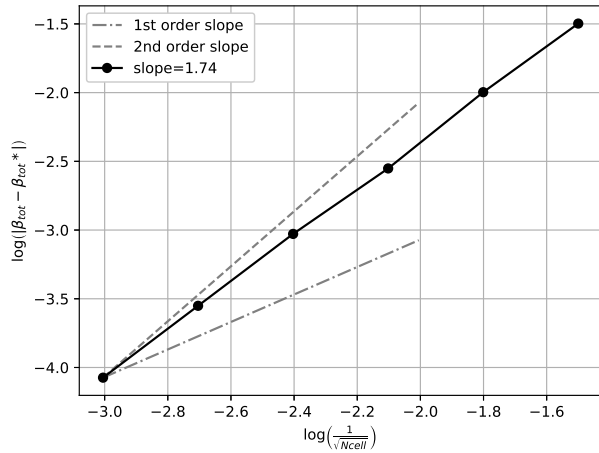


(b) Volume Penalization (VP)

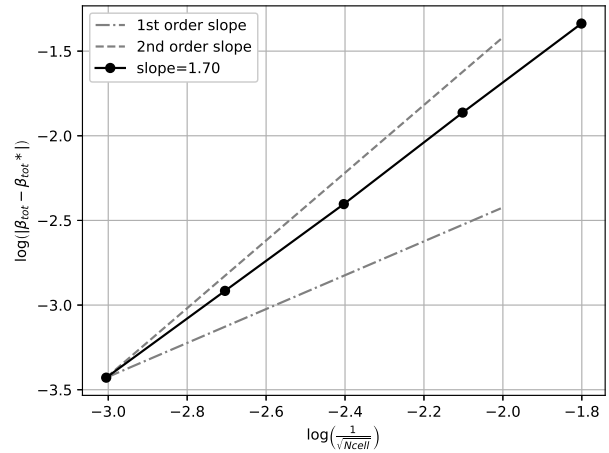


(c) Volume Penalization (VP-SSO)

Fig. 12 Order of convergence based on RMS_{ref}



(a) Body-Fitted



(b) Volume Penalization (VP)

Fig. 13 Order of convergence based on the β_{tot}

Globally, the IBMs are approaching 2^{nd} order accuracy similar to the body-fitted approach. This was not expected as the volume penalization method usually limits the order of accuracy to 1^{st} order. However, it is logical to obtain more than 1^{st} order accuracy for the droplet equations as no penalization is applied in the impingement zone, where β is evaluated. The penalization terms are only applied in the shadow zone where $\beta = 0$, thus not affecting the order of convergence.

C. NACA0012 in icing conditions

In this section, a clean NACA0012 airfoil is used to assess the penalization methods for typical icing conditions. This test case uses an unstructured mesh made of triangles with a far-field located at 50 chords (c). The wall cell size is $2.5e-3c$ with refinements at the leading edge (size $5e-4c$) and trailing edge (size $2.5e-4c$). A linear growth is applied from the wall to the far-field (size: $4.0c$) which gives a cell count of 77 172. For the immersed boundary mesh, the cell count is 112 172.

As the drag term is activated, the droplet trajectories are influenced by the aerodynamic solution. For this test case, it is evaluated using the penalized Euler equations [6] to deal with the immersed boundary. Using this approach, an aerodynamic solution is readily available in the solid ensuring continuity near the immersed boundary which is useful for the VP method. Note that the results presented in this section include the effect of both the penalized Euler [6] and droplet equations.

The wall pressure coefficients (C_p) are first compared against the body-fitted method to assess the quality of the aerodynamic solution, where a good match can be observed Fig. 14a. The comparison of the collection efficiency is presented in Fig. 14b where a good match can also be observed between the body-fitted and the penalization methods. Different mesh refinements were also tested for the NACA0012 case but only the mesh converged solution are shown here. Fig. 14b illustrates that with mesh refinement the BF and IB methods converge towards the same solution. Furthermore, smooth results are obtained for both the VP and VP-SSO methods. As a reminder, oscillations were observed with the VP-SSO on the cylinder case (Fig. 11b).

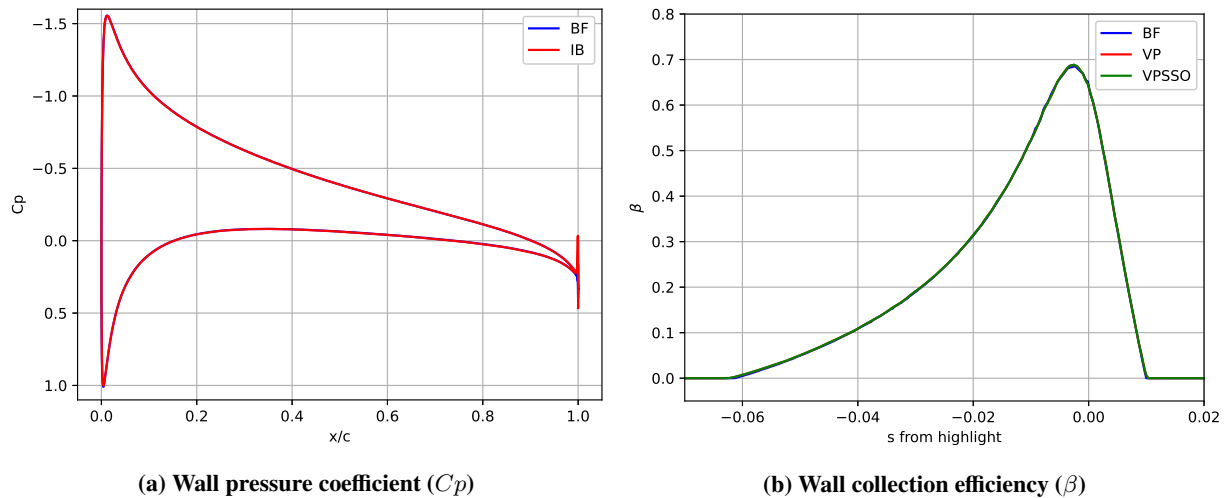


Fig. 14 Comparison of wall data for the NACA0012 airfoil

The α field is represented in Fig. 15 for the body-fitted and immersed boundary simulations. It shows that the combined mask function behaves correctly not only for the cylinder but also for the NACA0012 case. In the impingement zone, the droplets travel inside the solid but their re-injection in the fluid is avoided by the application of the Dirichlet condition in the shadow zone. Again there is a white cylinder inside the solid zone which acts as a dummy body in Fig. 15b. The immersed NACA0012 is represented in by the white line.

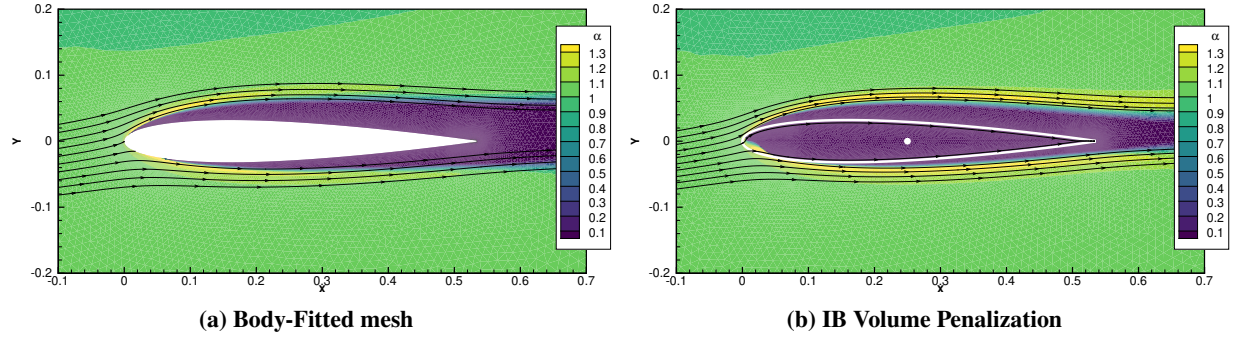


Fig. 15 Volume fraction of water (α) field

D. High Curvature Ice Horn Case

In this section an ice accreted GLC305 airfoil is used to assess the penalization method on a more challenging geometry (Fig. 16). This test case uses an unstructured mesh made of triangles with a far-field located at 20 chords (c). The wall cell size is $2.5e-3c$ with refinements at the leading edge (size $6.25e-4c$) and trailing edge (size $2.5e-3c$). A linear growth is applied from the wall to the far-field (size: $4.0c$).

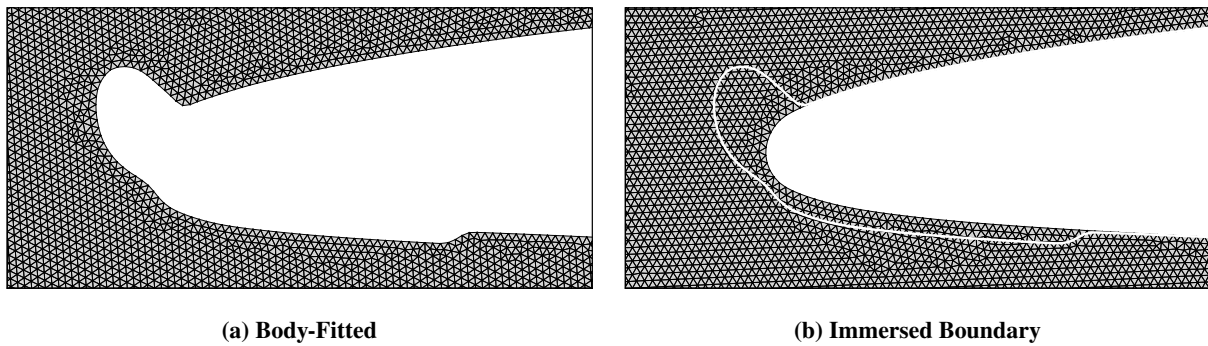


Fig. 16 Mesh around the ice horn

For this test case, the aerodynamic field is again evaluated using the penalized Euler equation of [6]. The pressure distribution C_p from Fig. 17a shows that a good match is obtained between the BF and IB methods. However the suction peak is slightly overestimated by the IB method (near $x/c = 0$). This does not seem to affect the collection efficiency (β) at the wall as shown on Fig. 17b, where the curves are overlaid for the BF, VP and VP-SSO methods. Again the collection efficiency is free of oscillation for both the VP and VP-SSO solution. This shows that the penalization method developed for the droplet equations are still able to reproduce the BF solution on airfoils and shows some potential for ice accreted airfoils.

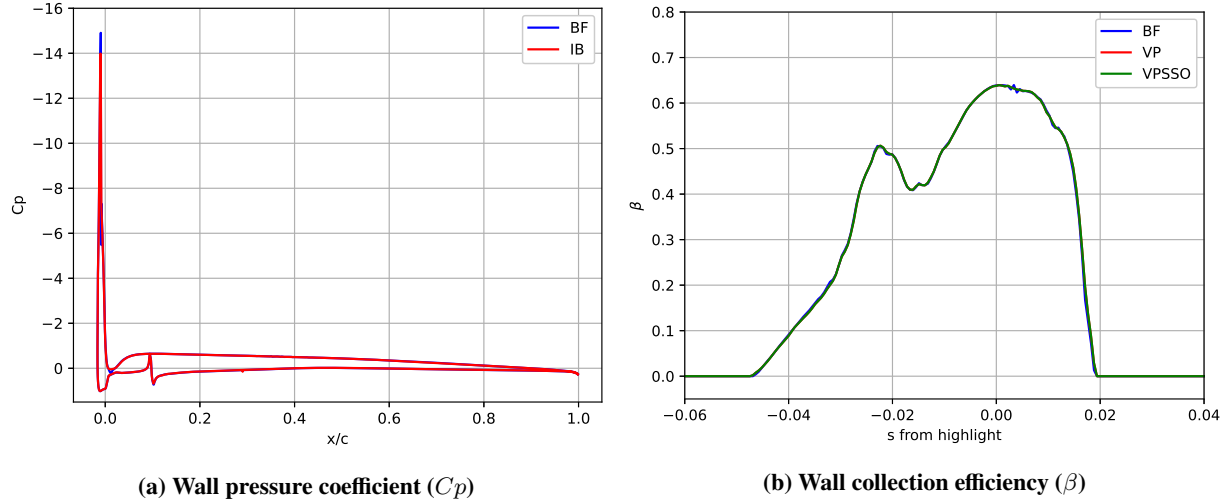


Fig. 17 Comparison of wall data for the ice horn

V. Conclusion

This paper proposes a penalization method for the Eulerian simulation of droplet impingement. The application of a penalization method is not straightforward for the droplet equations as the required wall boundary condition changes along the immersed boundary and must be treated with care to avoid that droplets enter the solid in an impingement zone and then be re-injected in the computational domain. A technique based on a double mask function is suggested to treat the wall boundary correctly.

Amongst the suggested penalization methods (VP and VP-SSO), a better behavior is achieved using the VP method, which is shown to reproduce the body-fitted solution on all cases while also providing a smooth solution. The activation of the physical source term in the solid zone (VP method) help in retrieving a smooth solution across the IB thus increasing the quality of the data extraction. However, it requires the definition of a fictitious aerodynamic field in the solid. On the other hand, the VP-SSO method is independent of the fictitious aerodynamics in the solid zone and is able to provide similar results to the VP and BF methods in terms of accuracy and smoothness for the clean NACA0012 airfoil and the ice horn. However, it generates an oscillating β distribution for the cylinder case (with non-zero droplet drag). With mesh convergence the oscillations become less significant and the collection efficiency tends towards the BF solution. These results suggest that the activation of the physical source term in the solid zone (VP method) is beneficial for some cases but is not always necessary.

The volume penalization (VP) is simple to implement, provide smooth results and is able to achieve second order accuracy like the body-fitted approach. Its application to multi-step ice accretion will be investigated further in the near future.

Appendix

With R_c the cylinder's radius, the analytical incompressible airflow solution around a cylinder in Cartesian coordinates is as follows:

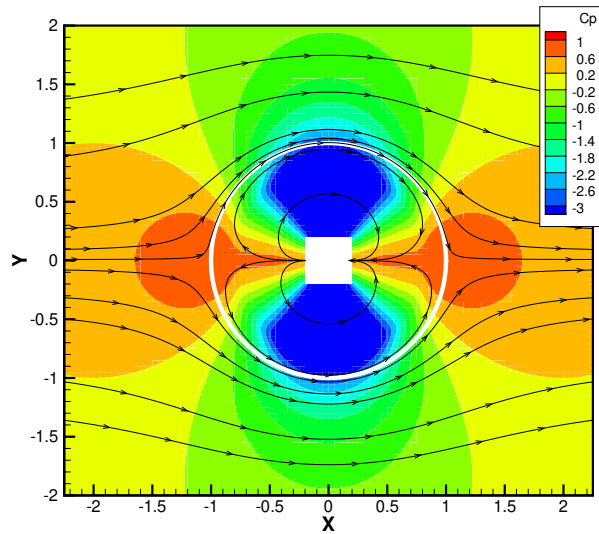


Fig. 18 Analytical aerodynamic field ($D/\Delta x = 40$)

$$\rho = \rho_\infty \quad (34)$$

$$u = U_\infty \left(1 + \frac{R^2}{r^4} (y^2 - x^2) \right) \quad (35)$$

$$v = -2U_\infty \frac{R^2}{r^4} xy \quad (36)$$

$$P = P_\infty + \frac{1}{2} \rho_\infty (U_\infty^2 - (u^2 + v^2)) \quad (37)$$

$$U_\infty = Mach \sqrt{\gamma \frac{P_\infty}{\rho_\infty}} \quad (38)$$

$$r = \sqrt{x^2 + y^2} \quad (39)$$

Acknowledgments

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), the Canada Research Chair program and the ONERA, the French Aerospace Lab.

References

- [1] Wright, W. B., Gent, R. W., and Guffond, D., "DRA/NASA/ONERA Collaboration on Icing Research: Part II—Prediction of Airfoil Ice Accretion," Tech. Rep. CR-202349, NASA, 1997. URL <https://ntrs.nasa.gov/citations/19970023937>.
- [2] Gent, R. W., Dart, N. P., and Cansdale, J. T., "Aircraft icing," *Philosophical Transactions of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, Vol. 358, No. 1776, 2000, pp. 2873–2911. <https://doi.org/10.1098/rsta.2000.0689>.
- [3] Wright, W. B., "User's Manual for LEWICE Version 3.2," CR 2008-214255, NASA, 2008. URL <https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/20080048307.pdf>.
- [4] Hasanzadeh, K., Laurendeau, E., and Paraschivoiu, I., "Quasi-Steady Convergence of Multistep Navier–Stokes Icing Simulations," *Journal of Aircraft*, Vol. 50, No. 4, 2013, pp. 1261–1274. <https://doi.org/10.2514/1.C032197>.
- [5] Trontin, P., Blanchard, G., Kontogiannis, A., and Villedieu, P., "Description and assessment of the new ONERA 2D icing suite IGLOO2D," *9th AIAA Atmospheric and Space Environments Conference*, 2017. <https://doi.org/10.2514/6.2017-3417>.
- [6] Lavoie, P., Radenac, E., Blanchard, G., Laurendeau, E., and Villedieu, P., "An Improved Characteristic Based Volume Penalization Method for the Euler Equations Towards Icing Applications," *Computer & Fluids*, 2020. , (Submitted).
- [7] Clarke, D. K., Hassan, H. A., and Salas, M. D., "Euler calculations for multielement airfoils using Cartesian grids," *AIAA Journal*, Vol. 24, No. 3, 1986, pp. 353–358. <https://doi.org/10.2514/3.9273>.
- [8] Berger, M., and Leveque, R., "An adaptive Cartesian mesh algorithm for the Euler equations in arbitrary geometries," *Fluid Dynamics and Co-located Conferences*, American Institute of Aeronautics and Astronautics, 1989, pp. –. <https://doi.org/10.2514/6.1989-1930>.
- [9] Berger, M., and Aftosmis, M., "Progress Towards a Cartesian Cut-Cell Method for Viscous Compressible Flow," *50th AIAA Aerospace Sciences Meeting*, 2012. <https://doi.org/doi.org/10.2514/6.2012-1301>, aIAA 2012-1301.

- [10] Mohd-Yusof, J., “Combined immersed-boundary/B-spline methods for simulations of flow in complex geometries,” *Annual Research Briefs, Center for Turbulence Research*, 1997, pp. 317–328. URL <https://ntrs.nasa.gov/archive/nasa/casi.ntrs.nasa.gov/19990063249.pdf>.
- [11] Mittal, R., Dong, H., Bozkurttas, M., Najjar, F., Vargas, A., and von Loebbecke, A., “A versatile sharp interface immersed boundary method for incompressible flows with complex boundaries,” *Journal of Computational Physics*, Vol. 227, No. 10, 2008, pp. 4825 – 4852. <https://doi.org/http://dx.doi.org/10.1016/j.jcp.2008.01.028>.
- [12] Fedkiw, R. P., Aslam, T., Merriman, B., and Osher, S., “A Non-oscillatory Eulerian Approach to Interfaces in Multimaterial Flows (the Ghost Fluid Method),” *Journal of Computational Physics*, Vol. 152, No. 2, 1999, pp. 457 – 492. <https://doi.org/http://dx.doi.org/10.1006/jcph.1999.6236>.
- [13] Peskin, C. S., “Flow patterns around heart valves: A numerical method,” *Journal of Computational Physics*, Vol. 10, No. 2, 1972, pp. 252–271. [https://doi.org/http://dx.doi.org/10.1016/0021-9991\(72\)90065-4](https://doi.org/http://dx.doi.org/10.1016/0021-9991(72)90065-4).
- [14] Angot, P., Bruneau, C.-H., and Fabrie, P., “A penalization method to take into account obstacles in incompressible viscous flows,” *Numerische Mathematik*, Vol. 81, No. 4, 1999, pp. 497–520. <https://doi.org/10.1007/s002110050401>.
- [15] Beaugendre, H., Morency, F., Gallizio, F., and Laurens, S., “Computation of Ice Shedding Trajectories Using Cartesian Grids, Penalization, and Level Sets,” *Model. Simul. Eng.*, Vol. 2011, 2011, pp. 3:1–3:15. <https://doi.org/10.1155/2011/274947>.
- [16] Liu, Q., and Vasilyev, O. V., “A Brinkman penalization method for compressible flows in complex geometries,” *Journal of Computational Physics*, Vol. 227, No. 2, 2007, pp. 946 – 966. <https://doi.org/http://dx.doi.org/10.1016/j.jcp.2007.07.037>.
- [17] Feireisl, E., Neustupa, J., and Stebel, J., “Convergence of a Brinkman-type penalization for compressible fluid flows,” *Journal of Differential Equations*, Vol. 250, No. 1, 2011, pp. 596 – 606. <https://doi.org/http://dx.doi.org/10.1016/j.jde.2010.09.031>.
- [18] Abgrall, R., Beaugendre, H., and Dobrzynski, C., “An immersed boundary method using unstructured anisotropic mesh adaptation combined with level-sets and penalization techniques,” *Journal of Computational Physics*, Vol. 257, 2014, pp. 83 – 101. <https://doi.org/http://dx.doi.org/10.1016/j.jcp.2013.08.052>.
- [19] Brown-Dymkoski, E., Kasimov, N., and Vasilyev, O. V., “A characteristic based volume penalization method for general evolution problems applied to compressible viscous flows,” *Journal of Computational Physics*, Vol. 262, 2014, pp. 344 – 357. <https://doi.org/https://doi.org/10.1016/j.jcp.2013.12.060>.
- [20] Brown-Dymkoski, E., Kasimov, N., and Vasilyev, O. V., “A Characteristic-Based Volume Penalization Method for Arbitrary Mach Flows Around Solid Obstacles,” *Direct and Large-Eddy Simulation IX*, edited by J. Fröhlich, H. Kuerten, B. J. Geurts, and V. Armenio. Springer International Publishing, Cham, 2015, pp. 109–115. https://doi.org/https://doi.org/10.1007/978-3-319-14448-1_15.
- [21] Capizzano, F., and Iuliano, E., “A Eulerian Method for Water Droplet Impingement by Means of an Immersed Boundary Technique,” *Journal of Fluids Engineering*, Vol. 136, No. 4, 2014, pp. 040906–040906–8. <https://doi.org/10.1115/1.4025867>.
- [22] Al-kebsi, A., Mose, R., and Hoarau, Y., “Multi-Step Ice Accretion Simulation Using the Level-Set Method,” *SAE International Conference on Icing of Aircraft, Engines, and Structure*, Minneapolis, United States, 2019. <https://doi.org/10.4271/2019-01-1955>, URL <https://hal.archives-ouvertes.fr/hal-02562783>.
- [23] Bourgault, Y., Habashi, W. G., Dompierre, J., and Baruzzi, G. S., “A finite element method study of Eulerian droplets impingement models,” *International Journal for Numerical Methods in Fluids*, Vol. 29, No. 4, 1999, pp. 429–449. [https://doi.org/https://doi.org/10.1002/\(SICI\)1097-0363\(19990228\)29:4<429::AID-FLD795>3.0.CO;2-F](https://doi.org/https://doi.org/10.1002/(SICI)1097-0363(19990228)29:4<429::AID-FLD795>3.0.CO;2-F).
- [24] Schiller, L., and Naumann, Z., “A drag coefficient correlation,” *Z. Ver. Deutsch. Ing.*, Vol. 77, 1935, pp. 318–323.
- [25] Durst, F., Miloievic, D., and Schönung, B., “Eulerian and Lagrangian predictions of particulate two-phase flows: a numerical study,” *Applied Mathematical Modelling*, Vol. 8, No. 2, 1984, pp. 101 – 115. [https://doi.org/https://doi.org/10.1016/0307-904X\(84\)90062-3](https://doi.org/https://doi.org/10.1016/0307-904X(84)90062-3).
- [26] Schneider, P. J., and Eberly, D. H., “Chapter 13 - computational geometry topics,” *Geometric Tools for Computer Graphics*, edited by D. H. SCHNEIDER, PHILIP J. and EBERLY, The Morgan Kaufmann Series in Computer Graphics, Morgan Kaufmann, San Francisco, 2003, pp. 673 – 825. <https://doi.org/https://doi.org/10.1016/B978-155860594-7/50016-3>.
- [27] Sarthou, A., Vincent, S., and Caltagirone, J., “A second-order curvilinear to Cartesian transformation of immersed interfaces and boundaries. Application to fictitious domains and multiphase flows,” *Computers & Fluids*, Vol. 46, No. 1, 2011, pp. 422 – 428. <https://doi.org/http://dx.doi.org/10.1016/j.compfluid.2010.11.008>, 10th ICFD Conference Series on Numerical Methods for Fluid Dynamics (ICFD 2010).

- [28] Etcheverlepo, A., “Développement de ,éthodes de domaines fictifs au second ordre,” Ph.D. thesis, Université Bordeaux I, 2013. URL http://ori-oai.u-bordeaux1.fr/pdf/2013/ETCHEVERLEPO_ADRIEN_2013.pdf.
- [29] Harten, A., Lax, P. D., and Leer, B. v., “On Upstream Differencing and Godunov-Type Schemes for Hyperbolic Conservation Laws,” *SIAM Review*, Vol. 25, No. 1, 1983, pp. 35–61. <https://doi.org/10.1137/1025002>.
- [30] Norde, E., “Eulerian method for ice crystal icing in turbofan engines,” Ph.D. thesis, University of Twente, May 2017. <https://doi.org/10.3990/1.9789036543392>.
- [31] Ladson, C. L., Brooks Jr., C. W., Hill, A. S., and Sproles, D. W., “Computer Program to Obtain Ordinates for NACA Airfoils,” Technical Memorandum 4741, NASA, Dec 1996. URL <https://ntrs.nasa.gov/search.jsp?R=19970008124>.
- [32] Vassberg, J. C., and Jameson, A., “In Pursuit of Grid Convergence for Two-Dimensional Euler Solutions,” *Journal of Aircraft*, Vol. 47, No. 4, 2010, pp. 1152–1166. <https://doi.org/10.2514/1.46737>.