

Multi-Step Ice Accretion Simulation Using The Level-Set Method

*A. Al-Kebisi**, *R. Mose** and *Y. Hoarau**
 *Strasbourg University, ICube, CNRS
 2 Rue Boussingault, 67100 Strasbourg, France
 aalkebsi@unistra.fr · hoarau@unistra.fr

Abstract

Ice accretion in aircraft structures is modelled using the Level-Set method. Due to the nature of the evolving ice shape and the re-meshing complexity, current icing models are limited. The Level-Set method allows fully multi-step simulation of ice accretion. The solid boundary is treated implicitly or explicitly. The solid body can be defined via a continuous characteristic level set function in the former case, and as a discrete layer or list of points in the later. Consequently, the boundary conditions for the airflow are applied through a penalization term, or by continuous forcing. On the other hand, the droplet transport is computed using an Eulerian approach in an Immersed Boundary Method coupled with Level-Set IBM-LS framework, in which the droplet fields impinge on an explicit layer of cells defined by the LS function using a discrete IBM formulation of the interface.

Introduction

Ice accretion reduces flight safety in extreme weather conditions. According to the US National Transport Safety Board, this is one of the main causes of flight accidents.²⁴ Icing frequently occurs when air vehicles passes through regions rich of suspended water droplets, which impinge on aircraft structures. These droplets either freeze on impact, get swept along the surface then freeze further from the point of impact, or do not freeze at all, depending on the temperature and other conditions. As a result, the designed aerodynamic surfaces are deformed by the introduction of new solid attached to the surface. The design of a reliable de-icing system requires a thorough knowledge of the icing phenomenon itself. Experimental icing study cannot exceed the scope of a handful of simple cases due to its complexity and cost. On the other hand, the use of numerical modelling makes it possible to simulate all possible configurations while studying a spectrum of different parameters.

Simple ice accretion codes in use, treat the icing process as a single layer in a quasi-steady state. A single icing step is assumed to be the result of four intermediate, quasi-steady, one-way coupled, consecutive steps:

1. Dry airflow is evaluated.
2. Suspended water droplets are transported by the air impacting against the surface.
3. A liquid film forms on the surface; the thermodynamic balance of this film is evaluated in order to determine the ice accretion rate.
4. The geometry is finally deformed by the accumulation of ice.

All these parameters, evaluated through steps one to three, are assumed not to change during the ice accumulation; only the flow parameters and variables of the initial clean geometry are used until the end of the ice accumulation.

The dry airflow simulation is usually evaluated using the two-dimensional panel method. More recently, several codes used by the industry use the more advanced Navier-Stokes solver, replacing the traditional panel method as in FENSAP-ICE (McGill University)⁴ or LEWICE3D.⁶

The droplet transport is modelled in either a Lagrangian or an Eulerian approach. In a Lagrangian representation, the droplet dynamics equation are derived from Newton's second law.²⁵ In an Eulerian representation however, a droplet concentration and momentum fields are transported. Following that, the impact rate is calculated geometrically in the Lagrangian formulation, and by direct determination in the Eulerian approach.²⁷ The Eulerian approach is more advantageous than the traditional Lagrangian approach, mainly due to its simplicity with multi-body and three-dimensional geometries and its direct integration to existing PDE based solvers.

The following step is to solve the mass and energy balance of the liquid layer accumulated on the surface by the impinging droplets. This is done in a quasi-steady state, where the impingement rate does not change anymore,

MULTI-STEP ICE ACCRETION

after being calculated in the previous step. The simplest method, called the Messinger model, is based on mass and energy balance.¹⁴ In the Messinger model, the liquid film is computed from the stagnation point and then going further downstream cell by cell towards the trailing edge. An improvement of this Method is the Messinger's iterative model proposed in,³² in which it is possible to treat multiple stagnation points and thus three-dimensional geometries. Other important improvements are based on the resolution of ordinary or partial differential equations where the ODE or PDE models the mass and energy conservation. Such methods were initially proposed by¹⁸ based on an ODE, and then extended for multi-body geometries by.²¹ This extension to the Messinger model added the possibility to solve an additional conduction heat transfer equation on the ice and water substrates, to better evaluate the ice accretion. Ultimately,⁷ proposed a PDE method, based on the Messinger model and the shallow water equation, abbreviated as SWIM. Such PDE model integrates perfectly within a Navier-Stokes air solver coupled with an Eulerian droplet model. The main result obtained from this step is the ice accumulation rate, represented as an ice thickness or an icing velocity.

In the last step, having calculated the ice thickness, the deformed geometry is obtained. Many techniques exist in the literature to obtain the deformed geometry^{8,11,16,30} such as grid movement techniques, or re-meshing.

A significant improvement on this single step ice assumption is to model the icing phenomenon as a multi-layer problem. Multi-layer ice accretion assumes the ice to be formed in quasi steady steps, layer-by-layer as a piecewise function of time. At the end of each layer, the flow field properties (steps 1-3) are reevaluated. Consequently, the mesh needs to be regenerated for each deformed ice shape, after each layer. Numerous studies showed that an improvement was achieved when using multi-step icing, notably for glaze ice configurations. However, applications are mainly restricted to two dimensional cases, since the automatic grid regeneration is nearly impossible without producing low quality grids. Only few three-dimensional cases are mentioned in the literature, and they are not well explained in terms of convergence and mesh quality, and are usually accompanied with poor heat transfer evaluation and poor ice formation compared to experimental results.

In this study, we develop an approach to completely avoid re-meshing. Such approach would overcome most of the limitations of re-meshing and most importantly would inspire Quasi-non-Steady ice accretion modelling. The core of such an approach is embedded grid techniques, namely Level-Set (LS)^{19,20} and Immersed Boundary Method (IBM).²³

In a Level-Set "LS" framework, the solid body is represented implicitly by a characteristic function. The Level-Set method was first used in icing framework to evaluate the ice shedding trajectories by.³ In this work, the level zero of the LS function represents the evolving ice/air interface. Also, the inside zone could be used to represent the accreted ice and thus can be used to evaluate the heat transfer in this substrate, allowing the study of deicing in a coupled transient system. Likewise, the outside zone represents the external airflow zone. By refining close to the wall, a high quality flow can be obtained. The mesh in question can start from a body-fitted mesh or from any mesh since the grid is embedded and distance normal to the wall is not necessarily conditioned. Such embedded-grid method integrates perfectly with the Eulerian formulation of PDEs, overcoming meshing issues. Embedded-grids can be performed using structured grids.^{2,12,15} The no slip boundary condition at the wall can be achieved through a penalty term, in which the solid is treated as an impermeable medium in the so called Brinckman-Navier-Stokes equations.

In previous papers,^{10,22} we developed and validated an icing simulation code in the NSMB flow solver. The Ice solver consisted of four modules: compressible air solver, Eulerian droplet solver, SWIM solver, ALE grid regeneration module. The compressible Navier-Stokes air solver is now implemented with a penalization term to reproduce the no slip boundary condition at the wall. The Eulerian droplet module is integrated with an IBM-LS formulation where the impingement boundary condition is applied to an explicit representation of the solid wall. Currently, we are limited to rime ice configurations, since the third module is still under development for an embedded grid framework. These different modules are integrated within the Level-Set framework, where the zero level represents the advancement of the iced surface. The Level-Set equations are discretised using a fifth order spatial WENO scheme and a third order time Runge-Kutta scheme to ensure consistency. The LS is redistanced after each ice layer by solving a reinitialization equation. The icing velocity evaluated at the interface is propagated throughout the whole domain by solving an additional PDE. The implementation of turbulence wall laws to model the turbulent boundary layer is still under progress. All the modules were developed for multi-block grids parallelised with the MPI environment. Chimera superimposed grids are supported as well since they are used to ensure fine grids close to the wall.

Numerical method and governing equations

The ice accretion modules are developed in the NSMB Solver (Navier-Stokes Multi-block solver).³¹ The NSMB solver utilizes structured, multi-block, chimera grids, supports grid motion, contains parallel compressible and incompressible NS solvers and is able to treat embedded grid techniques such as IBM and LS.

Penalized compressible flow

In the airflow solver step the solid body is defined implicitly by the LS function. In such implicit representation or LS disposition the solid is namely given by a characteristic function χ_s , which is set to a smoothed Heaviside function H . The smoothed Heaviside function is defined using the signed distance LS function in our model $H(-\phi)$. Choosing the LS function as a signed distance enhances the numerical accuracy.^{17,19} A penalty is added as a source term to the NS equations to enforce the no-slip boundary condition at the wall.^{1,2} This term resembles the continuous Boundary Condition forcing used in IBM. The only difference is that the Heaviside function in LS framework, is constructed from the LS signed distance function, whereas in pure IBM it is constructed from the Lagrangian points defining the interface. The penalized compressible NS system takes the form seen in Eq. 1

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) &= 0 \\ \frac{\partial \rho \mathbf{u}}{\partial t} + \nabla \cdot (\rho \mathbf{u} \otimes \mathbf{u}) &= \nabla \cdot \underline{\pi} + f_u \\ \frac{\partial \rho e}{\partial t} + \nabla \cdot ((\rho e + p) \mathbf{u}) &= \nabla \cdot (\underline{\pi} \mathbf{u} + \mathbf{q}) + f_e \end{aligned} \quad (1)$$

$$f_u = \frac{1}{\eta} \chi_s (\rho \mathbf{u} - \rho \mathbf{u}_s)$$

$$f_e = \frac{1}{\eta} \chi_s \theta_s \rho_a (\epsilon(T) - \epsilon(T_s)) + (\rho \mathbf{u} - \rho \mathbf{u}_s) \cdot \mathbf{u}$$

The variable ρ represents air density, \mathbf{u} the dry airflow velocity vector, e the airflow specific total energy, p the pressure, \mathbf{q} the heat flux, and $\underline{\pi}$ the stress tensor. The new additional terms f_u and f_e represent the penalty terms for the momentum and energy equations respectively, where $\frac{1}{\eta}$ is a penalization parameter, and χ_s is the characteristic function of the solid. As mentioned earlier, the characteristic function is a smoothed Heaviside function computed from the signed distance LS function, both of which are given in Eq. 2,3

$$H(\phi) = \begin{cases} 0 & \phi < -\epsilon \\ 0.5 \left(1 + \frac{\phi}{\epsilon} + \frac{\sin(\pi\phi/\epsilon)}{\pi} \right) & |\phi| \leq \epsilon \\ 1 & \phi > \epsilon \end{cases} \quad (2)$$

$$\chi_s = H(-\phi) \quad (3)$$

Eulerian Droplet Transport

The droplet transport module developed in NSMB implements an Eulerian approach.¹⁰ Wherein, a system of conservation equations of the non-dimensionalized droplet velocity and volume fraction are solved. The droplets are assumed to have a spherical shape, which flattens under high Reynolds numbers. Furthermore, the size is assumed to have a (LANGMUIR "D") distribution. Readers could refer to¹⁰ where all the other assumptions are justified, or to²⁷ first to propose the method. The governing conservation equations are given in Eq. 4

$$\left\{ \begin{aligned} \frac{\partial \alpha}{\partial t} + \nabla \cdot (\alpha \mathbf{u}_d) &= 0 \\ \frac{\partial \alpha \mathbf{u}_d}{\partial t} + \nabla \cdot (\alpha \mathbf{u}_d \otimes \mathbf{u}_d) &= \mathbf{F}_D + \mathbf{F}_{BG} \\ \mathbf{F}_D &= \alpha \frac{C_D Re_d}{24K} (\mathbf{u}_a - \mathbf{u}_d) \\ \mathbf{F}_{BG} &= \alpha \left(1 - \frac{\rho}{\rho_w} \right) \frac{1}{Fr^2} \mathbf{g} \end{aligned} \right. \quad (4)$$

The variable α represents the non-dimensionalized water volume fraction, \mathbf{u}_d non-dimensionalized water droplet velocity field, \mathbf{u}_a non-dimensionalized air velocity field, ρ_w water density, and \mathbf{g} gravity vector. The term $K = \rho d U_\infty / 18 L \mu$ is an inertia parameter, L represents the reference length, and d is the droplet median diameter. Re_d , the droplets Reynolds number for a spherical particle relative to the air flow phase is defined based on the slip velocity as in Eq. 5.

$$Re_d = \frac{\rho d |\mathbf{u}_a - \mathbf{u}_d|}{\mu_a} \quad (5)$$

The variable C_D is the drag coefficient of the droplets and is defined empirically as a function of the droplets shape. For spherical droplets we use the form proposed by.²⁶ For super large droplets (SLD), the droplets are assumed to deform

MULTI-STEP ICE ACCRETION

into discs. This results in a modification of the drag coefficient. A more elaborate, shape dependent, drag coefficient takes the form in equation 6

$$C_d = (C_{d,sphere} + ee(C_{d,disk} - C_{d,sphere})) \times Re_d \quad (6)$$

The variable ee is used as a weight factor and is given by another empirical equation 7. It represents a measure of the deformation of the droplet from a sphere to a disk. $C_{d,sphere}$ is the drag coefficient of a sphere, and $C_{d,disk}$ that of a disk.

$$ee = 1 - (1 + 0.007We^{0.5})^{-6} \quad (7)$$

The variable We is the Weber number which measures the relative importance of the droplet's inertia to its surface tension. It is given by equation 8.

$$We = \rho_d |\mathbf{u}_a - \mathbf{u}_w|^2 D / \sigma \quad (8)$$

The variable σ is the droplets surface tension approximated at 0.0756 N/m for water.

Since we assume that all droplets are captured by the surface and since the model treats the icing in quasi-steady state, the droplets boundary condition requires special treatment. During the droplet Eulerian transport step, droplets are assumed to disappear on the solid body in zones where they impact against the solid body, to simulate an a steady state impingement. In other words, the solid acts as a Neumann outlet. However, in dry zones where droplets do not impact against the solid body, the solid should act as a Dirichlet BC. The droplet velocity vector is checked at the interface and the appropriate BC is applied locally, accordingly. This method applied in¹³ is shown in equation 9

$$\begin{cases} if \mathbf{u} \cdot \mathbf{n} < \mathbf{0} & Wet \rightarrow Neuman & \frac{\partial u}{\partial n} = 0, \frac{\partial \alpha}{\partial n} = 0 \\ if \mathbf{u} \cdot \mathbf{n} > \mathbf{0} & Dry \rightarrow Dirichlet & \mathbf{u} = \mathbf{0}, \alpha = \mathbf{0} \end{cases} \quad (9)$$

In a Level-Set framework this adaptive boundary condition cannot be applied implicitly, no continuous penalty term can achieve such behavior, at least to our knowledge. An explicit forcing of the boundary condition using an IBM representation of the LS implicit function can easily achieve this adaptive BC. Consequently, we use the first layer of cells adjacent to the interface to directly enforce the required boundary condition on the discretized form of the Eulerian droplet equations.

IBM-LS Technique: Droplet BC

Consider Fig. 1. The adaptive BC is imposed on the cells marked with x and are henceforth called immersed boundary IB points. The procedure is summarized in the following points:

- (1) First we detect the first layer of points outside the interface by detecting the change of $sign(\phi)$ with neighbors. An array containing their indices is saved IBp_{ind} .
- (2) Each point is back mirrored against the solid surface resulting in an image point (im) further from the original point at a distance Δ from the IB point along the normal direction. The normal direction is available from the signed-distance LS function $\mathbf{n} = \nabla\phi$. The distance Δ is calculated from the cell size to ensure that the image point (im) is close to a fluid cell, and outside the IB cell.
- (3) The coordinates of these image points are calculated via Eq. 10 and are saved in an array \vec{X}_{im} .
- (4) The distance to the wall at these image points which is the same as the LS function is calculated via Eq. 11 and saved in a third array ϕ_{im} .
- (5) An algorithm searches for the fluid cell closest to the image point. The indices of the closest point are saved in an array CP_{ind} .

- (6) This closest point is used to interpolate the state vector \mathbf{Q} to the image point. The gradient of the droplet state vector is calculated at our list of closest points $\nabla \mathbf{Q}_{cp}$ and is then used to interpolate the state vector \mathbf{Q}_{cp} to the image point as given in Eq. 12.
- (7) The required BC is imposed by checking the droplet velocity component normal to the wall. Consequently, we impose $\mathbf{Q}_{IB} = \mathbf{Q}_{ip}$ for the wet impingement zones where $\mathbf{u}_w \cdot \mathbf{n} < 0$ and $\mathbf{Q}_{IB} = \mathbf{Q}_{ip} * \phi_{ip} / \phi_{im}$ for the dry shaded zones where $\mathbf{u}_w \cdot \mathbf{n} > 0$.
- (8) If and only if the Level-Set is moved by advection, resulting in a new interface, the arrays IBP_{ind} , \vec{X}_{im} , ϕ_{im} , and CP_{ind} are re-evaluated.

$$\vec{X}_{im} = \vec{X}_{wp} + \nabla \phi_{wp} * \Delta \quad (10)$$

$$\phi_{im} = \phi_{wp} + \Delta \quad (11)$$

$$\mathbf{Q}_{ip} = \mathbf{Q}_{cp} - \nabla \mathbf{Q}_{cp} \cdot (\Delta \cdot \nabla \phi) \quad (12)$$

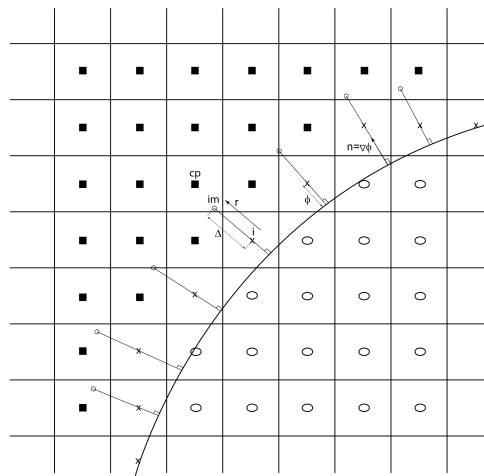


Figure 1: Schematics of IBM-LS framework. ■: flow domain, x: IB cells layer, ○: solid domain.

This method was also developed using two image points. Using two image points allows higher order interpolation consequently allowing enforcing a zero gradient to better apply the Neumann BC on the wall. Another variation of the method was also developed, where we reconstruct the flux at the wall instead of at the cell center as proposed by.⁹ However, we do not follow the method depicted in,⁹ where he extrapolates from the image point to the wall without enforcing a zero gradient. We found that the first variation presented here was more stable than the other two.

Collection Efficiency

The collection efficiency is directly calculated using the formula given in Eq. 13.

$$\beta = \alpha \mathbf{u} \cdot \mathbf{n} \quad (13)$$

In the IBM-LS configuration this variable can be calculated at either the first layer of cells outside which are accessible through the array WP_{ind} , or at the cells whose Dirac Delta $\delta(\phi)$ is higher than zero and whose LS function ϕ is positive.

Thermodynamic modelling of the liquid film

The thermodynamic model originally developed in NSMB is based on the work of,⁷⁵ called the Shallow-Water for Ice Modelling SWIM. The SWIM module is based on the shallow-water equation, basically equations of conservation of mass and energy. The interested reader can refer to¹⁰ where the implementation of the model is fully explained. In an embedded grid framework, we are currently still developing the appropriate variation of the model. Consequently, only dry rime ice configurations are being studied so far, where the impinging mass is assumed to solidify on impact. Thus, resultant mass rate of ice accretion \dot{m}_{ice} is evaluated through Eq. 14.

$$\dot{m}_{ice} = \dot{m}_{imp} = LWC \cdot V_{\infty} \cdot \beta \quad (14)$$

The variable LWC represents the liquid water content and V_{∞} the free stream or far fields velocity.

MULTI-STEP ICE ACCRETION

Level-Set Function

The Level-Set approach was developed in¹⁰ to track the ice air interface evolution for a single-step ice accretion model. The ice/air interface is represented implicitly by the zero iso-contour of the Level-Set function ϕ . The LS function ϕ is set as a signed distance function.¹⁹ A signed distance function is given by: $|\phi(\vec{x})| = d(\vec{x})$. It is initialized in the computational domain as follows:

- $\phi = d$, in the outside zone (air)
- $\phi = -d$, in the inside zone (ice,solid)
- $\Gamma = x|\phi(\mathbf{x}, t) = 0$, the interface

The variable d is the distance to the interface Γ .

The LS approach replaces the need for an initial solid with body fitted mesh and thus the remeshing after each icing shot. However, one can still start from an initial solid with a body fitted mesh; in such case, the negative part of the LS function, the inside zone, can be used to define the ice substrate where it is possible to solve an additional heat transfer equation. The accumulation of ice resulting in deformation of the solid/air interface is achieved by advecting the LS function "the LS equation". Consequently, The Level-Set advection is written in Eq. 15.

$$\frac{\partial \phi}{\partial t} + \vec{v}_{ice} \cdot \nabla \phi = 0 \quad (15)$$

The variable \vec{v}_{ice} represents the LS velocity vector. This velocity is equal to the ice accretion velocity on the interface. The velocity at the interface is calculated via Eq. 16.

$$\vec{v}_{wall} = \frac{\dot{m}_{ice}}{\rho_{ice}} \cdot \nabla \phi \quad (16)$$

This velocity calculated only at the interface needs to be propagated in the normal direction to be defined in the whole domain and then used to advect the LS function ϕ . An important assumption made here is that ice accumulates in the direction normal to the solid. To propagate the icing velocity in the normal direction an additional PDE system is solved and is given in Eq. 17

$$\frac{\partial w}{\partial t} + sign(\phi)(\vec{\nabla} \phi) \cdot \nabla w = 0 \quad (17)$$

The variable w represents the evolution of the propagation of \mathbf{v}_{wall} normal to the wall. The solution of Eq. 17 converges to the steady-state required Level-Set velocity $w_{steady} = \vec{v}_{ice}$. The resulting velocity field is then used in Eq. 15 to advance the geometry by advecting the LS function ϕ .

When the LS function is changed (after each ice layer), the signed distance feature of the LS function ϕ is not confirmed any more. During a single ice-step while advecting the LS function, the signed distance feature can get deteriorated based on the grid's coarseness and the discretization methods used. Reinitializing ϕ while being advected can restore the values around the interface to signed distance to reduce numerical dissipation. Advecting a signed distance function is numerically stable and ameliorates conservation. Reinitializing ϕ at the end of each ice layer is essentially required to calculate the new solid characteristic function χ_s from the smoothed Heaviside function $H(\phi)$. To recover $|\Phi(\vec{x})| = d(\vec{x})$ we solve the reinitialization equation Eq. 18 as proposed by.¹⁹ This equation can be rewritten as shown in Eq. 19 in a conservative finite volume representation.

$$\frac{\partial \phi}{\partial t} + |\nabla \phi| = 1 \quad (18)$$

$$\frac{\partial \phi}{\partial t} + sign(\phi) \frac{\nabla \phi}{|\nabla \phi|} \cdot \nabla \phi = sign(\phi) \quad (19)$$

Where the $sign(\phi)$ is approximated numerically by the smooth function in Eq.20.

$$sign(\phi) = \frac{\phi}{\sqrt{\phi^2 + |\nabla \phi|^2 \Delta x^2}} \quad (20)$$

It was advised that Eq. 19 does not respect local mass conservation at the interface. In other words, while re-distancing ϕ in the domain the cells in which the zero interface passes do not conserve the parts at each sides of the interface. The distance being calculated perfectly all around the interface, does not guarantee that the interface would not move locally in the cells through which it passes. In ice accretion simulation, the ice mass or icing rate is the most critical

variable. An alteration of the local amount of ice recurring each time ϕ is re-distanced would make the whole approach lose its appeal. Many methods exist in the literature to locally preserve the interface. We chose to use the method advised by,²⁸ which was initially proposed by,²⁹ and given in Eq. 21

$$\frac{\partial \phi}{\partial t} + \text{sign}(\phi_0) \frac{\nabla \phi}{|\nabla \phi|} \cdot \nabla \phi = \text{sign}(\phi) + \lambda \delta(\phi_0) |\nabla \phi_0| \quad (21)$$

The variable λ represents the new Correction Factor used to enforce local mass conservation and is given in Eq. 22. $\delta(\phi_0)$ on the other hand, is the smoothed Dirac Delta function and is given in Eq. 23

$$\lambda = - \frac{\int_{\Omega_{i,j,k}} \delta(\phi_0) (\phi^{n+1} - \phi_0) / \Delta t}{\int_{\Omega_{i,j,k}} \delta^2(\phi_0) |\nabla \phi_0|} \quad (22)$$

$$\delta(\phi) = \begin{cases} 0 & \phi < -\epsilon \\ 0.5 (1 + \sin(\pi\phi/\epsilon)) & |\phi| \leq \epsilon \\ 1 & \phi > \epsilon \end{cases} \quad (23)$$

Both integrals in Eq.22 are integrations over the cell $\Omega_{i,j,k}$ and are easily evaluated from a 9-point stencil in 2d Eq. 24 or from a 27-point stencil in 3d as given in Eq. 25

$$\int_{\Omega_{i,j}} f = \frac{\Delta x^2}{24} \left(16f_{i,j} + \sum_{m,n=-1;(m,n) \neq (0,0)}^1 f_{i+m,j+n} \right) \quad (24)$$

$$\int_{\Omega_{i,j,k}} f = \frac{\Delta x^3}{78} \left(52f_{i,j,k} + \sum_{m,n,q=-1;(m,n,q) \neq (0,0,0)}^1 f_{i+m,j+n,k+q} \right) \quad (25)$$

Eq. 19 is first solved as a prediction step then Eq. 21 corrects ϕ . The additional source term impacts the LS function ϕ only at the interface where $\delta(\phi) > 0$. consequently, one can avoid the computational cost of the correction step by calculating the correction term only at points where $\delta(\phi) > 0$.

The LS equations system (15,17,18) are all discretized using a finite volume formulated TVD WENO5 spatial scheme and a RK3 temporal scheme. To limit communication time in during parallel communications in the MPI environment, only two ghost cells are communicated between neighboring blocks, and only two ghost cells are interpolated and communicated for chimera blocks. Thus, the WENO5 scheme falls to a third order TVD scheme with a flux limiter on the block connectivity when the 5 point stencil required for WENO5 is not present. This is because the WENO5 stencil needs three neighbouring cells, whereas the TVD scheme used uses only two neighbouring cells.

The final algorithm is shown in Fig. 2.

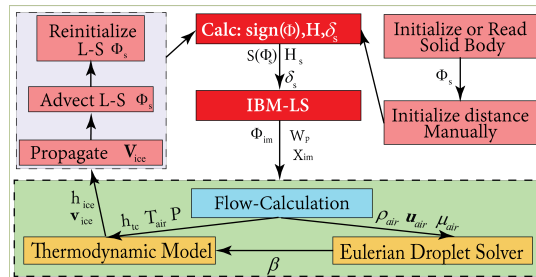


Figure 2: IBM-LS multi-layer ice accretion algorithm diagram.

Results

In this section, icing simulations are performed on two-dimensional NACA0012 airfoil. The flow field is solved explicitly. The flow can be laminar or turbulent; however, wall laws for the LS embedded grids are still under development. We only present rime ice configurations. Three Cartesian meshes have been generated. The coarse mesh contains 318×288 cells in a single block and the size of cells close to the wall is 0.008, the second mesh 1300×1000 with cell size 0.002 and in 64 blocks, and the third mesh a chimera mesh 16800×16800 with cell size 0.0006 and in 128 blocks.

MULTI-STEP ICE ACCRETION

Level-Set Equations

The signed distance function is first initialized to -1 inside and $+1$ outside. The distance is then manually initialized in a small band around the interface. Without this manual initialization, the interface is highly deformed. The manual initialization compares the distance between a cell and all the points used to generate the NACA0012 profile. Using this well initialized ϕ we solve the reinitialization equation 21. If we use the equation 19 without the correction term the interface gets deteriorated as seen in figure 3 in red. It is clear that all cells keep their sidedness relative the interface, but only cells through which the interface passes do not conserve quantities on both sides, inducing mass loss. The optimized result obtained with equation 21 is seen in the same figure in blue which coincides exactly with the initial black interface hardly recognizable since they perfectly match, indicating perfect local conservation.

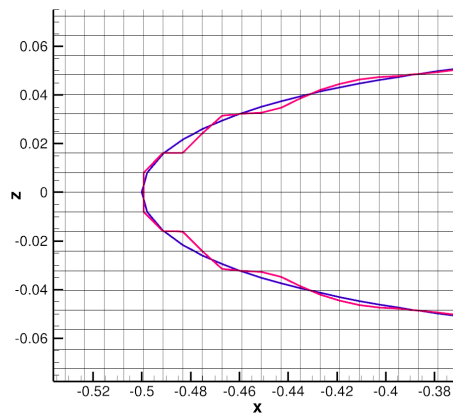


Figure 3: Re-distancing the well initialized ϕ with "blue" and without "red" local conservation.

Reinitialization of the LS function ϕ is necessarily carried out after each icing layer. For conditional stability, the time step depends on the cell sizes since the temporal scheme is a RK3 explicit scheme. Since the advecting velocity is the gradient of ϕ which has a magnitude of $|\nabla\phi| = 1$, setting the time step equal to the smallest cell size was satisfactory, thus reducing the cost of calculating the flux and the resulting CFL at all the cells. The equation is solved a number of iterations enough to ensure that the zone susceptible to icing is well covered. Thus, the number of iterations needed can be calculated from the icing velocity and the exposure time. In multi-block grids, only blocks close to the wall are treated; in other words, all the LS equations (15,17,19) are solved only for grid blocks close to the wall where icing is anticipated. All of these equations are only solved until the zone susceptible to icing is well covered. The velocity \mathbf{v}_{ice} propagation normal to the wall achieved by equation 17 is shown in figures 4 over the clean geometry for the coarsest grid. The second figure Fig. 5 shows the propagated \mathbf{v}_{ice} and the resulting LS function ϕ advected by this field \mathbf{v}_{ice} .

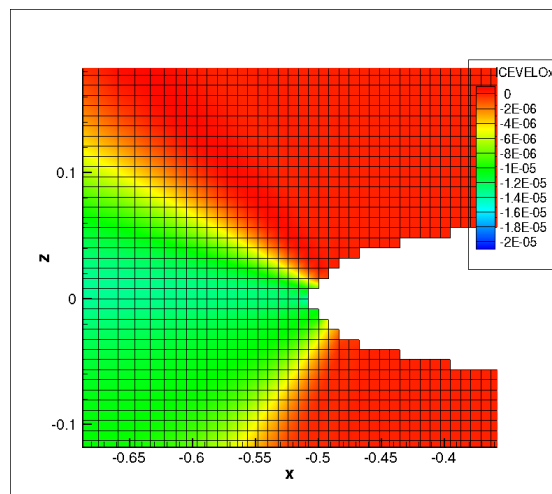


Figure 4: The x component of \mathbf{v}_{ice} propagated normal to the wall for the coarsest grid.

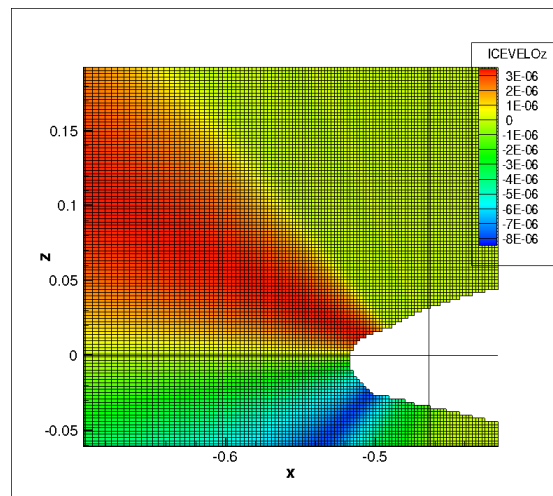


Figure 5: The z component of \mathbf{v}_{ice} propagated normal to the wall for the second multi-block grid and the resulting ϕ .

In Fig. 6 we show the deformed geometry after four ice shots each of five minutes exposure. The contours represent the x component of the icing velocity propagated normal to the wall, and the streamlines follow the droplet trajectories, that clearly impinge on the wall.

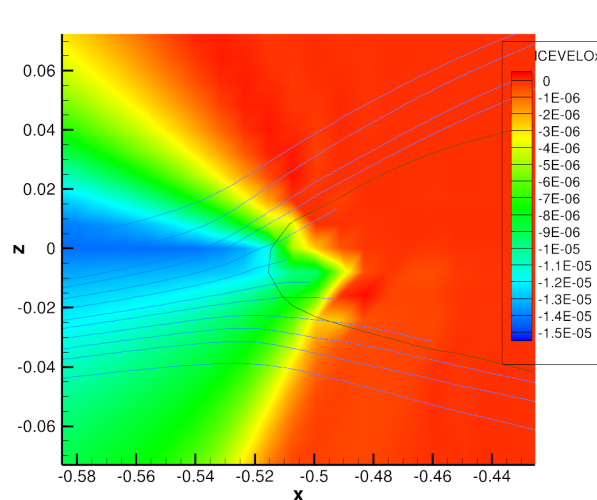


Figure 6: 4th shot iced geometry, resulting x component of \mathbf{v}_{ice} contours, droplet trajectories streamlines.

In Fig. 7 and 8 we show the deformed geometry obtained for the fine and coarse grids using 1, 2, and 5 ice layers.

Conclusion

A multi step icing model using the Level-Set methodology has been presented in detail and introduced as a main to achieve unsteady transient ice accretion simulation. This approach was developed in the NSMB "Navier Stokes Multi-Block structured finite volume compressible flow solver.

The icing is assumed to result in quasi steady steps, layer by layer. It is computed on an immersed grid utilizing both the Level-Set capabilities as well as Immersed Boundary Method, each when convenient. The air flow is evaluated using the compressible Navier Stokes equation. We currently restrict our calculations to laminar flows; turbulent models like the Spallart-Allmaras model are currently without adaptive wall laws. The solid body is present through its characteristic equation and thus we use continuous forcing to reproduce the no-slip boundary condition. Droplets tracking are computed implicitly using first order spatial scheme. The SWIM model is currently not applicable in the LS framework. The impinging mass is assumed to freeze on impact resulting in rime ice. The mass rate is evaluated from which we evaluate the ice accretion velocity, that is propagated normal to the wall in the whole domain. The Level-Set

MULTI-STEP ICE ACCRETION

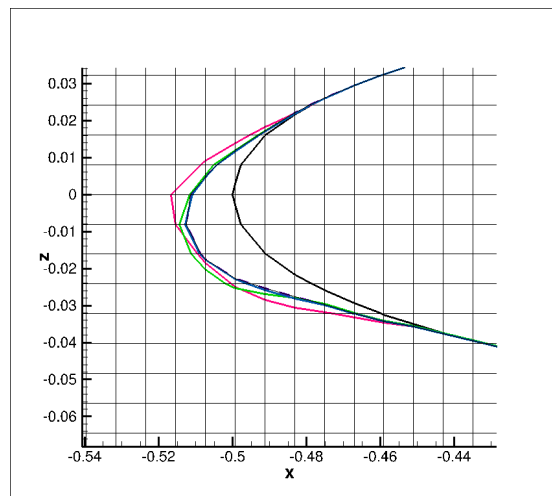


Figure 7: Deformed Geometry (adverted ϕ) obtained on the coarse mesh using 1,2,5 and 10 "coinciding with blue" ice shots.

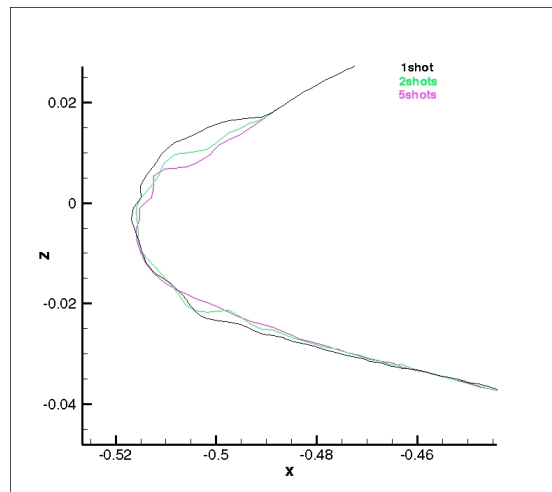


Figure 8: Deformed Geometry (adverted ϕ) obtained on the second fine mesh using 1,2 and 5 ice shots.

equation is used to advect the wall. After each layer the Level-Set is redistanced, with local mass conservation taken into account. The Level-Set equations are discretized using the WENO5 spatial scheme, and the RK3 temporal scheme. Everything is parallelized using the MPI environment.

References

- [1] R. Abgrall, H. Beaugendre, and C. Dobrzynski. An immersed boundary method using unstructured anisotropic mesh adaptation combined with level-sets and penalization techniques. *Journal of Computational Physics*, 257:83–101, 2014.
- [2] P. Angot, C.-H. Bruneau, and P. Fabrie. A penalization method to take into account obstacles in viscous flows. *Numerische Mathematik*, 81:497–520, 02 1999.
- [3] H. Beaugendre, F. Morency, F. Gallizio, and S. Laurens. Computation of ice shedding trajectories using cartesian grids, penalization, and level sets. *Modelling and Simulation in Engineering*, 2011:3, 2011.
- [4] H. Beaugendre, F. Morency, and W. G. Habashi. Development of a second generation in-flight icing simulation code. *Journal of fluids engineering*, 128(2):378â387, 2006.

- [5] H. Beaugendre, F. Morency, and W.G. Habashi. FENSAP-ICE's three-dimensional in-flight ice accretion module: ICE3D. *Journal of Aircraft*, 40(2):239â247, 2003.
- [6] Colin S Bidwell and Mark G Potapczuk. Users manual for the NASA Lewis three-dimensional ice accretion code (LEWICE 3D). *Technical Memorandum*, 105974, 1993.
- [7] Y. Bourgault, H. Beaugendre, and W. Habashi. Development of a shallow-water icing model in fensap-ice. *Journal of Aircraft*, 37(4):640 â 646, 2000.
- [8] Y. Cao, J. Huang, and J. Yin. Numerical simulation of three-dimensional ice accretion on an aircraft wing. *International Journal of Heat and Mass Transfer*, 92:34–54, 2016.
- [9] Francesco Capizzano and Emiliano Iuliano. A eulerian method for water droplet impingement by means of an immersed boundary technique. *J. Fluids Eng*, 2014.
- [10] E. Laurendeau D. Pena and Y. Hoarau. A single step ice accretion model using level-set method. *Journal of Fluids and Structures*, 65:278–294, 2016.
- [11] K. Hasanzadeh, E. Laurendeau, F. Saeed, and I. Paraschivoiu. Wing aerodynamic performance analysis and stall prediction using canice2d-ns icing code. In *20th Annual Conference of the CFD Society of Canada.*, 2012.
- [12] X.Y. Hu, B.C. Khoo, N.A. Adams, and Huang F.L. A conservative interface method for compressible flows. *Journal of Computational Physics*, 219:553–578, 2006.
- [13] S. Jung and R. Myong. Numerical modeling for eulerian droplet impingement in supercooled large droplet conditions. *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, January 2013.
- [14] Bernard L Messinger. Equilibrium temperature of an unheated icing surface as a function of air speed. *Journal of the Aeronautical Sciences (Institute of the Aeronautical Sciences)*, 20(1), 1953.
- [15] R. Mittal and G. Iaccarino. Immersed boundary methods. 37(1):239–261, 2005.
- [16] E. Montreuil, A. Chazottes, D. Guffond, A. Murrone, F. Caminade, and S. Catris. ECLIPPS: 1. three-dimensional CFD prediction of the ice accretion. *1st AIAA Atmospheric and Space Environments Conference*, 2009.
- [17] W. Mulder, S. Osher, and Sethian J. Computing interface motion in compressible gas dynamics. *Journal of Computational Physics*, 100:209–228, 1992.
- [18] Tim G. Myers. Extension to the Messinger model for aircraft icing. *AIAA journal*, 39(2):211â218, 2001.
- [19] S. Osher and R. Fedkiw. *Level set methods and dynamic implicit surfaces*, volume 153. Springer Science & Business Media, 2006.
- [20] S. Osher and J.A. Sethian. *Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations.*, volume 79. 1988.
- [21] S. Ozgen and M. Canibek. Ice accretion simulation on multi-element airfoils using extended messingermodel. *Heat and Mass Transfer*, 45:305 – 322, 2009.
- [22] D. Pena, Y. Hoarau, and E. Laurendeau. Development of a three-dimensional icing simulation code in the NSMB flow solver. *International Journal of Engineering Systems Modelling and Simulation.*, 8(2):86–98, 2015.
- [23] C. S. Peskin. Flow patterns around heart valves: A numerical. *Journal of Computational Physics*, 10(2):252â271, 1972.
- [24] Al Reehorst. National transportation safety board aircraft accident investigation supported. 2005.
- [25] W. Sang, Y. Shi, and C. Xi. Numerical simulation of icing effect and ice accretion on three-dimensional configurations. *Science China Technological Sciences*, 56(9):2278–2288, September 2013.
- [26] L Schiller and Z Naumann. A drag coefficient correlation. *Vdi Zeitung*, 77(318):51, 1935.
- [27] J.N. Scott, W.L. Hankey, F.J. Giessler, and T.P. Gielda. Navier-stokes solution to the flowfield over ice accretion shapes. *Journal of Aircraft*, 25(8):710–716, 1988.

MULTI-STEP ICE ACCRETION

- [28] Z. Solomenko, P.D.M. Spelt, L. Ó Náraigh, and P. Alix. Mass conservation and reduction of parasitic interfacial waves in level-set methods for the numerical simulation of two-phase flows: A comparative study. *International Journal of Multiphase Flow*, 95:235–256, 2017.
- [29] M. Sussman and E. Fatemi. An efficient interface preserving level set redistancing algorithm and its application to interfacial incompressible fluid flow. *SIAM Journal of Scientific Computing*, 20(4):1165–1191, 1999.
- [30] P. Verdin, J. P. F. Charpin, and Thompson C. P. Multistep results in icecremo2. *Journal of Aircraft*, 46(5):1607–1613, 2009.
- [31] JB Vos, AW Rizzi, A Corjon, E Chaput, and E Soinne. Recent Advances in Aerodynamics inside the NSMB (navier-stokes multi-block) Consortium. *AIAA paper*, 802:1998, 1998.
- [32] C. Zhu, B. Fu, Z. Sun, and C. Zhu. 3D ice accretion simulation for complex configuration basing on improved messinger model. *International Journal of Modern Physics: Conference Series*, 19:341–350, January 2012.