On the use of a 2D Finite-Volume Integral Boundary Layer Method for Ice Accretion Calculations

Emmanuel Radenac, Charlotte Bayeux, Philippe Villedieu

On the use of a 2D Finite-Volume Integral Boundary Layer Method for Ice Accretion Calculations

Emmanuel Radenac[1] and Charlotte Bayeux[2] and Philippe Villedieu[3]

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

In this paper, a two-dimensional integral boundary layer method developed in a recent work is applied to ice accretion computations. The method has already been validated in terms of boundary layer dynamic effects in another article. It is here validated for its ability to capture ice shapes, once the method is included in an icing suite. To be more specific, results using the new boundary layer method are compared against experimental ice shapes and simulated ones with the widely-used simplified integral method. The validation is carried out at an aggregated level because icing databases generally provide access to final ice shapes only. But since the simplified integral method is used in many icing numerical tools, this comparison makes it possible to investigate the benefits of introducing the new method for calculating the boundary layer. The main outcome of the new method is an improvement of the prediction of the boundary layer prediction under smooth-wall assumption, which in turn improves ice shape prediction. It is shown that, overall, the ice shapes are indeed either better predicted with the new method than with the baseline approach, or equally predicted with both methods. In addition, since the heat transfer coefficient tends to be underestimated by simplified integral methods, the new approach tends to predict lower horn angles than the baseline approach. Finally, the consequences of these results on current and future developments of ice accretion solvers are discussed. In particular, the new method is better suited to a 3D extension than the simplified integral method.

———

[1] Research engineer, Aerodynamics and Energetic Modeling Dept., BP 4025 2 avenue Ed. Belin, Toulouse, France.
[2] Research engineer, Airbus Defence and Space, 31 rue des Cosmonautes, Toulouse, France.
[3] Research director, Aerodynamics and Energetic Modeling Dept., BP 4025 2 avenue Ed. Belin, Toulouse, France.

Nomenclature

2D   bi-dimensional

3D   three-dimensional

AOA angle of attack (deg.)

c chord length (m)

C_D viscous dissipation coefficient

C_f friction coefficient

CFL Courant-Friedrichs-Lewy number

c_p heat capacity at constant pressure (J/K/kg)

C_p pressure coefficient

$\boldsymbol{F}$ flux vector

H shape factor

h_{tc} heat transfer coefficient (W/K/m^2)

k_s equivalent sand-grain roughness height (m)

LWC Liquid Water Content (kg/m^3)

MVD Median Volume Diameter (m)

P pressure (Pa)

Pr   Prandtl number

$\boldsymbol{S}$ source term vector

St   Stanton number

t icing exposure time (s)

T temperature (K)

$\boldsymbol{U}$ vector of primary variables

u velocity (m/s)

x, y Cartesian coordinates (m)

β collection efficiency

δ boundary-layer thickness (m)

δ_1 displacement thickness (m)

δ_3 kinetic-energy thickness (m)

θ momentum thickness (m)

ν kinematic viscosity (m^2/s)

ρ density (kg/m^3)

τ shear stress (N/m^2)

$<>$ Reynolds averaging

$()_d$ quantity linked to droplets

$()_e$ boundary layer edge quantity

$()_s$ smooth-wall quantity

$()_t$ turbulent quantity

$()_w$ wall quantity

$()_\infty$ farfield condition

$()^+$ quantity in wall coordinates, $()^+ = \dfrac{() \times u_e \sqrt{C_f/2}}{\nu_e}$

I.   Introduction

Aircraft in-flight icing prediction is one field for which it has remained common practice to calculate viscous flow by coupling inviscid fluid and boundary layer codes. Indeed, for both certification and aircraft design, a large number of icing conditions must be calculated within a reasonable time. In addition, the calculation procedure is sequential and quasi-steady. As the ice grows, the shape of the wing or obstacle is updated. The flow needs to be systematically calculated around this updated form. For a given shape, many aerodynamic calculations may therefore be necessary. As a result, a robust and fast calculation method for viscous flows is interesting for in-flight icing applications. This is especially true in 3D because the grids are obviously still larger than in 2D.

Integral boundary layer models are often employed in 2D icing suites (LEWICE [1], ONICE [2], IGLOO [3]). They are based on the integration of the boundary layer equations in the direction normal to the wall. This integration results in the loss of one space dimension and thus in significant

saving in computational time. As far as icing simulations are concerned, the choice of such a method (instead of using a full Navier-Stokes solver) is justified by the fact that the aerodynamic velocity field in the boundary layer is not useful to compute the droplet collection efficiency and the ice accretion growth rate. Only the values of the skin friction and the heat exchange coefficients must be provided by the air flow solver to the accretion solver [4] . The ability of the boundary layer solver to accurately compute separated flows is not mandatory either. Indeed ice accretion generally occurs in the leading edge area (region of highest droplet collection efficiency) where the boundary layer is attached and, when separation occurs due to the ice deposit itself, the heat exchange coefficient in the separated zone is much lower than in the attached part and hence does not need to be accurately predicted.

For these reasons, the development of a general and robust 3D integral boundary layer Finite-Volume solver, able to deal with unstructured surface meshes, seems to be a promising way to provide aeronautic industry with efficient tools for icing applications. This opinion also relies on the fact that today, even for 2D computations, many existing icing tools (LEWICE, ONICE, IGLOO for instance) use simplified integral boundary layer models, such as the well-known method of Thwaites [5], and nevertheless are able to provide satisfactory results for most icing conditions. However, the 3D extension of these simplified boundary layer models is not straightforward since the formulation on which they are based is only two-dimensional. More general models and numerical methods, which could be formulated both in 2D and in 3D, are necessary. This was done in a previous paper [6], a work in continuity with many other studies performed during the last decades to extend the application domain of integral boundary-layer models [7–14]. A 2D model, based on a Finite Volume resolution and easy to extend to 3D, was indeed derived in [6]. The ability of the model to accurately capture boundary layer quantities was also evaluated, as well as its robustness on complex 2D ice shapes [6]. However, no application of this model for computing ice growth was performed. It was thus deemed necessary to demonstrate that this model was interesting for calculating ice shapes. This is the purpose of the present article, where we still restrict our attention to the 2D case. More specifically, the aim is to validate the use of the new model in an icing suite against experimental ice shapes. As ice-accretion calculations are often made with a simplified integral method, the interest

of this article is also to assess and analyze the contribution of the new method in comparison with the basic method.

The paper is divided into three parts. In the first one, the baseline ice accretion method (used in IGLOO2D [3]) is recalled. In particular, for the boundary layer, it is based on the simplified integral method. In the second part, the integral boundary layer model developed in [6] is briefly described. In a third part, two ice accretion databases are investigated with both the baseline approach and the new model. The accuracy of the new model is thus assessed by comparing the results in terms of ice shapes.

II.   Baseline approach for ice accretion computations

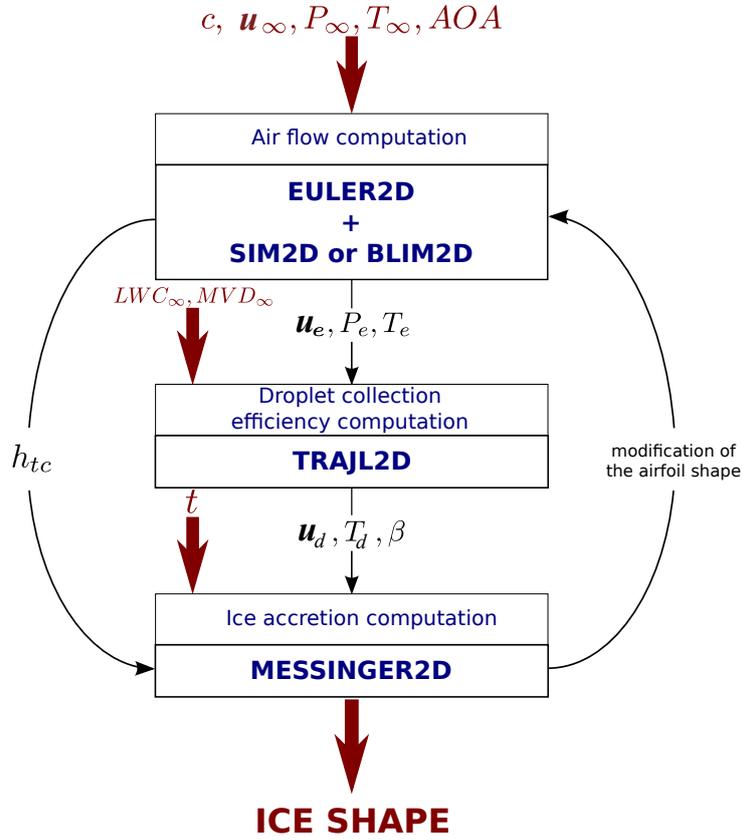### A.   Overall sequential process

Fig. 1: *IGLOO2D flow-chart*

The computation of ice accretion on an airfoil (or any other icing surface) is generally made

through a sequential approach (figure 1):

1. calculation of the airflow: an inviscid approach is often employed, like in IGLOO2D [3]. This step is mandatory to provide the carrying flow to the supercooled droplets and to give the pressure distribution to the ice accretion code. Furthermore, it provides the edge velocity and pressure distribution to the integral boundary layer code. Here the solver EULER2D of IGLOO2D was used.

2. computation of droplet trajectories: this can be done either in a Lagrangian or in a Eulerian way. The collection efficiency $\beta = \dot{m}_{imp}/(LWC_\infty|u_\infty|)$ produced by the simulation is a key parameter for the ice accretion computation ($\dot{m}_{imp}$ is the mass flow rate of the impinging droplets on the airfoil, LWC_∞ is the farfield liquid water content and u_∞ is the farfield air velocity). The Lagrangian approach (TRAJL2D, see appendix A 2) was used for the present article because it is routinely used in IGLOO2D (especially because the complex droplet-wall interaction models required for supercooled large droplets and ice crystals are easier to implement than in a Eulerian code and the computational cost is not prohibitive on 2D grids).

3. computation of the boundary layer: an integral boundary layer method is often employed. The key outputs are the heat transfer and skin friction coefficients, which are major inputs of the ice accretion calculation. The heat transfer coefficient has a particularly important role for glaze ice shapes because water solidification and runback are conditioned by convective heat transfer. As presented subsequently, two solvers will be compared in this paper (SIM2D and BLIM2D).

4. ice accretion simulation: the Messinger model is generally employed, for this step. It is based on the steady-state calculation of mass and energy balance on the iced surface. The main output of the code (here MESSINGER2D, see appendix A 1) is the local ice accretion rate, which can be easily converted into ice thickness for a given exposure time.

The process can be repeated several times to update the ice shape. Several kinds of grids can be used, structured or unstructured.

In the present paper, the IGLOO2D icing suite will be employed. The baseline quick approach,

which is mainly used in this paper, is based on the use of structured grids with 128 points on the iced airfoil (see figure 2 for an example of such a grid, see appendix A 3 for a description of the grid generation method) and the use of the so-called "predictor-corrector" approach. This method consists of updating the ice shape only once. A first "predictor" stage indeed produces an ice shape from the computation on the clean airfoil. In the "corrector" stage, the aerodynamic and droplet flows are computed around the predicted ice shape. The Messinger balance is finally performed by interpolating the aerodynamic and droplet inputs between those of the predictor calculation and the corrector calculation.

It is also possible to use finer unstructured grids and to perform multiple successive computations in a multi-step approach (also called multi-shot or multi-layer methods in other icing suites of the literature). In this alternative approach, the overall exposure time is divided into N time steps. N "predictor" computations are thus successively performed on the updated grids (corresponding to the ice shape at the given iteration). The interested reader will find further information in the article of Trontin [3] for multi-step and predictor-corrector approaches in IGLOO2D, as well as in [1, 15, 16] for multi-step approaches in other codes.

The main approach will thus consist in using the predictor-corrector approach of IGLOO2D on 128-points structured grids. Among the IGLOO2D solvers, no specific work was done on EULER2D, TRAJL2D, SIM2D and MESSINGER2D for this article. Only BLIM2D was specifically developed in the article of Bayeux [6] and inserted in IGLOO2D for this article, as an alternative to the fairly standard SIM2D code. BLIM2D is thus at the heart of the article. It will be described in section III, whereas SIM2D will be described in the following section.

B. Boundary layer simplified integral method

For the boundary layer, a simplified method is often used in icing suites. In IGLOO2D, the baseline solver for this method is SIM2D [3]. It means that simplified algebraic relations are used to obtain the integral thicknesses, such as the momentum and energy thicknesses, for several regimes (laminar or turbulent regime, on smooth or rough walls). For instance, in fully turbulent regime, the momentum thickness is locally computed through the following relation, which was derived under

7

zero-pressure-gradient flat-plate and smooth-wall assumptions [3]:

$$\theta^{5/4} = \frac{0.016}{u_e(s)^{4.11}} \int_0^s u_e(x)^{3.86} \nu_e^{1/4} \mathrm{d}x \tag{1}$$

where s is the wrap distance and θ is the incompressible momentum thickness:

$$\theta = \int_0^\infty \frac{u}{u_e} \left(1 - \frac{u}{u_e}\right) \mathrm{d}y \tag{2}$$

u is the streamwise velocity in the boundary layer whereas u_e is the edge velocity, which is an input of the calculation. ν_e is the air kinematic viscosity.

For ice accretion calculations, the key output of the integral method is the heat transfer coefficient h_{tc}. It can be related to a Stanton number St, through u_e, the air density ρ and the air heat capacity at constant pressure c_p:

$$h_{tc} = \rho u_e c_p St \tag{3}$$

There are two distinct regions, a purely laminar region followed by a purely turbulent region:

- a laminar region, in the vicinity of the stagnation point, where strictly no impact of wall roughness is expected. The Smith and Spalding correlation is employed (Pr is the Prandtl number of air) [17]:

$$St = \frac{\nu_e \sqrt{u_e(s)^{2.87}}}{u_e(s) Pr \sqrt{11.68 \int_0^s \nu_e u_e(x)^{1.87} \mathrm{d}x}} \tag{4}$$

- once a transition criterion is met, the boundary layer abruptly becomes turbulent and rough-wall conditions are assumed as well. This is questionable, since, for instance, Kerho evidenced that a quite large transitional area exists on iced surfaces [18]. However, this simplified approach is the baseline of many icing suites [1, 3, 4]. In the rough-wall turbulent area, the Makkonen model [19] is used, which was derived after the work of Kays and Crawford [20]. In most ice accretion suites [1, 3, 4, 21], the following formulas are thus employed for skin friction and heat transfer coefficient, possibly including some slight adjustments:

$$St = \frac{C_f/2}{\mathcal{P}r_{ts} + \sqrt{C_f/2}/St_k} \tag{5}$$

where:

$$\frac{C_f}{2} = \frac{0.1681}{\ln^2 \left[864 \frac{\theta}{k_s} + 2.568 \right]} \tag{6}$$

and:

$$St_k = 1.92 k_s^{+ -0.45} \cdot \mathcal{P}r^{-0.8} \tag{7}$$

For air, $\mathcal{P}r_{ts} = 0.9$ and $\mathcal{P}r = 0.7$. Most parameters are mainly defined from the edge flow (u_e, ρ, c_p). Since these parameters are inputs of the integral boundary layer computation, the model is thus dependent on the momentum thickness θ and the equivalent sand-grain roughness height k_s only. Equation 1 provides θ. It must be noticed that smooth-wall assumption is made for equation 1. The computation of θ on a rough wall precludes the use of a simplified integral method. Consequently, model 5 is generally fed with a smooth-wall momentum thickness, which is not so obvious from a theoretical perspective, but seems to be rather effective [22].

Although it is questionable after the work of Kerho [18], a widely used transition criterion assumes a critical Reynolds number $\mathrm{Re}_{k,c} = 600$. In IGLOO2D, the criterion is adapted in the following manner: transition occurs once

$$\frac{k_s u_e}{\nu_e} \geq \mathrm{Re}_{k,c}. \tag{8}$$

In practice, a rough estimate of k_s is often used. In IGLOO2D, for an airfoil,

$$k_s = c/1000 \tag{9}$$

where c is the chord. This is obviously a rough estimate of the roughness height, as the roughness height is expected to depend on many more parameters than a single length scale of the iced area (as shown in references [23, 24] for instance). It is therefore a known limitation of the method. However, comparing equation 9 against the databases of the literature shows that this often provides a correct order of magnitude for k_s, although the trends of roughness spatial and temporal variations are obviously not captured by this approach. Since there are no proven models that can accurately reproduce both the transition over iced profiles and the roughness characteristics of real ice, equations 8 and 9 will be used here. The article focuses on the impact of a more general integral method for the boundary layer resolution.

III. Integral Boundary Layer model and its Finite Volume resolution

For the sake of completeness, the model and the Finite Volume approach proposed in reference [6] are briefly presented in this section. This method was implemented in the code BLIM2D. This solver is part of the icing suite IGLOO2D and can be employed instead of SIM2D.

In this method, a system of two equations is solved in order to compute the dynamic behavior of the boundary layer [6]. In addition to the momentum thickness defined in equation 2, these equations involve incompressible integral quantities which are defined as:

$$\delta_1 = \int_0^\infty \left(1 - \frac{u}{u_e}\right)\,\mathrm{d}y$$

$$\delta_3 = \int_0^\infty \frac{u}{u_e}\left(1 - \frac{u^2}{u_e{}^2}\right)\,\mathrm{d}y$$

δ_1 and δ_3 correspond to the displacement and kinetic-energy thicknesses respectively. The equations are unsteady, put under conservation form and assume incompressible flow hypothesis:

$$\frac{\partial \boldsymbol{U}}{\partial t} + \frac{\partial \boldsymbol{F}(x, \boldsymbol{U})}{\partial x} = \boldsymbol{S}(x, \boldsymbol{U}) \tag{10}$$

where

$$\boldsymbol{U} = \begin{pmatrix} u_e\delta_1 \\ u_e^2\theta \end{pmatrix} = \begin{pmatrix} U_1 \\ U_2 \end{pmatrix} \quad \boldsymbol{F} = \begin{pmatrix} u_e^2\theta \\ u_e^3\delta_3 - u_e^3\theta \end{pmatrix} \quad \boldsymbol{S} = \begin{pmatrix} -u_e\delta_1\dfrac{\partial u_e}{\partial x} + \dfrac{1}{2}u_e^2 C_f \\ (u_e^2\delta_1 - u_e^2\theta)\dfrac{\partial u_e}{\partial x} - \dfrac{1}{2}u_e^3 C_f + 2u_e^3 C_D \end{pmatrix}$$

C_f and C_D denote respectively the friction coefficient and the dissipation coefficient:

$$\frac{1}{2}u_e{}^2 C_f = \frac{\tau_w}{\rho}, \qquad u_e{}^3 C_D = \frac{1}{\rho}\int_0^\infty \tau\frac{\partial u}{\partial y}\,\mathrm{d}y$$

where $\dfrac{\tau}{\rho} = \nu\dfrac{\partial u}{\partial y}$ in laminar flow regime and $\dfrac{\tau}{\rho} = \nu\dfrac{\partial u}{\partial y} - <u'v'>$ in turbulent flow regime. τ_w is the skin friction. In equation 10, the first equation is an unsteady version of the well-known von Kármán equation. The second one is the kinetic energy integral equation.

The conservative formulation allows the use of a Finite Volume resolution. The resolution method is intended for unstructured grids and a cell-centered Finite Volume approach is used, as shown in more detail in Bayeux's article [6]. A first-order basic upwind scheme is used as follows. Let the numerical flux $\boldsymbol{G}_{i+1/2}$ be an approximation of the flux vector $\boldsymbol{F}$ at the node $x_{i+1/2}$ of edge

i. If $u_{e_{i+1/2}} > 0$:

$$\boldsymbol{G}_{i+1/2} = \begin{pmatrix} u_{e_{i+1/2}}^2 \theta_i \\ u_{e_{i+1/2}}^3 (\delta_{3i} - \theta_i) \end{pmatrix}$$

Otherwise:

$$\boldsymbol{G}_{i+1/2} = \begin{pmatrix} u_{e_{i+1/2}}^2 \theta_{i+1} \\ u_{e_{i+1/2}}^3 (\delta_{3i+1} - \theta_{i+1}) \end{pmatrix}$$

The system is solved by a semi-implicit approach with Euler explicit integration of numerical fluxes $\boldsymbol{F}$ and implicit integration of source terms $\boldsymbol{S}$. The unsteady resolution makes it possible to reach steady-state without specifying the location of the stagnation point. The two primary variables U_1 and U_2 are linked to the displacement and momentum thicknesses, respectively. Equation 10 involves three other terms, δ_3, C_f and C_D, which must be related to the solved variables. Some closure relations must thus be defined. Assumed velocity profiles and empirical relations are employed to provide the closure relations as well as self-similar boundary-layer theoretical results. The interested reader will find further information in the article of Bayeux [6]. Note that the proposed closure relations are valid for smooth walls only.

There are several key advantages of this method, compared to the simplified method used in SIM2D:

- there are fewer simplifying assumptions. For instance, in equation 1, θ comes from flat-plate assumptions, which are not necessary in the closure relations used for equation 10 [6].

- it can be extended to 3D. There are indeed no intrinsic limitations to the theory, as long as representative closure relations are found (whereas, on the contrary, transverse flows in the boundary layer cannot be accounted for in the simplified approach). Moreover, the equations are put in a conservative form, suitable for Finite Volume methods, which can be efficiently used on unstructured grids (suitable for the discretization of complex 3D shapes).

Regarding the thermal boundary layer, the method employed in the present paper to obtain the heat transfer coefficient was to use equation 4 in the laminar area and equations 5 and 6, fed with the smooth-wall value of the momentum thickness θ, computed thanks to model 10, in the turbulent

11

region. This is basically the same approach as in SIM2D but with a more accurate computation of the dynamic boundary layer and hence of the smooth-wall momentum thickness θ.

IV.   Comparison between BLIM2D and SIM2D

In this section, some IGLOO2D computations are performed on two databases presented in section IV A. For these simulations, the baseline approach of IGLOO2D was mainly used: the predictor-corrector approach was employed with structured grids, EULER2D was used for the airflow solution, TRAJL2D for the calculation of the droplet trajectories and MESSINGER2D was used for the ice accretion computation. BLIM2D is compared against SIM2D for the boundary layer computation, which allows to see the interest of the new code for the computation of the heat transfer (section IV C) and for the ice shape prediction (section IV D).

A.   Investigated databases

The impact of the new integral method on the ice accretion calculation was investigated on the database studied in the benchmark of reference [3]. The simulated icing conditions are displayed in table 1. This database was investigated by Trontin et al [3] to compare the results of IGLOO2D against the experimental ice shapes on the one hand and against LEWICE2D on the other hand [25]. The predictor-corrector approach employed in the present article and a multi-step approach were compared in reference [3]. The role of the grid size was also assessed in reference [3]. For some cases, the multi-step approach was more successful than the predictor-corrector approach in capturing the experimental ice shape. But overall, the predictor-corrector method and the use of a coarse structured mesh proved satisfactory. These same conclusions will be confirmed in section IV B . This justifies the use of this method here in sections IV C and IV D, especially since the calculations are then less expensive and there is one less parameter to adjust (the number of steps).

Database 1 is a set of cases for which ice accretion suites generally provide moderate agreement with experiment. Another database investigated by Trontin et al [3] was also computed to provide a set of conditions for which the agreement between the experimental ice shapes and the baseline IGLOO2D computations (using SIM2D) is already satisfactory. The icing conditions corresponding

Table 1: *Icing conditions for ice shape benchmark of icing suites by Trontin et al [3].*

Condition No.	Airfoil	Chord m	Airspeed m/s	AoA deg.	MVD μm	LWC g/m^3	Static Temp. ^oC	Spray Time s
1	NACA0012	0.3048	67.9	3	40	1.00	-7.1	900
2	NACA0012	0.3556	77.2	0	175	0.99	-12.3	678
3	NACA0012	0.5334	102.9	4	30	1.80	-10.9	360
4	NACA0012	0.9144	77.1	0	15	1.00	-11.3	1224
5	NACA23012	0.4572	102.9	2	15.4	0.75	-7.5	300
6	NACA23012	1.8288	90.0	2	15	0.94	-9.0	600
7	GLC305	0.3048	139.4	0	19	1.12	-17.6	138
8	GLC305	0.4572	88.4	0	29	1.47	-10.0	240
9	GLC305	0.6096	89.4	0	35	1.30	-10.0	360
10	GLC305	0.6096	89.4	0	43	1.27	-17.0	360
11	GLC305	0.9144	90.0	6	20.0	0.54	-4.8	1350
12	NLF0414	0.9144	66.9	2	20.0	0.54	-5.3	1350
13	LTHS	0.9144	130.2	0	21.0	0.56	-9.8	1476
14	NACA0015	0.3531	95.2	0	19	0.75	-10.0	600

to the second database are displayed in table 2.

The two databases address a wide range of conditions. In particular, the flow velocity was varied in a range from around 67 m/s to 140 m/s, whereas the airfoil chord is in the range from 0.30 m to 1.83 m. The Reynolds number $u_e c / \nu_e$ is thus in the range from $1.6 \cdot 10^6$ to $1.3 \cdot 10^7$. Several airfoil shapes were also investigated and the angle of attack is in the range from 0 to 6 degrees. In addition, both rime ice cases (cases A,B) and glaze ice cases were investigated. Of course, there are more glaze ice cases because the convective heat transfer (and thus the boundary layer effect) does not affect the rime ice shapes. Finally, there are cases with short or long exposure time t to icing conditions (cases 4, 11, 12, 13 especially for long spray time). For large t, developed ice shapes are expected, which could lead to large separation zones downstream the ice horns. The article [6]

Table 2: *Icing conditions for ice shape validation database by Trontin et al [3].*

Condition No.	Airfoil	Chord m	Airspeed m/s	AoA deg.	MVD μm	LWC g/m^3	Static Temp. $^\circ C$	Spray Time s
A	NACA0012	0.5334	103.15	4	20	0.55	-22.45	420
B	GLC305	0.9144	128.37	1.5	20	0.31	-15.55	348
C	NACA0012	0.5334	102.91	4	20	1.	-10.85	231
D	NACA0012	0.5334	102.91	4	15	0.6	-10.85	384
E	GLC305	0.9144	128.23	1.5	15	0.5	-9.55	432
F	GLC305	0.9144	89.64	6	20	0.54	-4.85	720

showed that the boundary layer computations are more difficult in these conditions. First, there are several stagnation points. Second, some backflow occurs, which leads to the presence of points where two external flows converge at the horn tips (as shown in figure 2). BLIM2D was shown to be robust to these issues by Bayeux et al [6], which is confirmed by the set of computations in conditions of tables 1 and 2.

B. Effect of the grid and of the time strategy

Reference [3] showed that IGLOO2D simulations with a 128-point structured mesh (as described in section A 3) and the predictor-corrector approach are generally satisfactory compared to simulations with an unstructured grid, even a refined one, and a multi-step approach. This demonstration was of course performed using the solver SIM2D. The same approach is reproduced here with the solver BLIM2D.

First, the effect of the mesh is shown. Several unstructured grids composed of triangles were generated with GMSH [26]. Figure 3 shows the meshes used for the computations of case 9. The 128-point structured grid is superimposed on the same figure. Different levels of refinement were used, as shown in table 3. Figure 3 shows that the coarse unstructured mesh is much coarser than the 128-point structured grid at the leading edge (whereas it gradually becomes finer than the
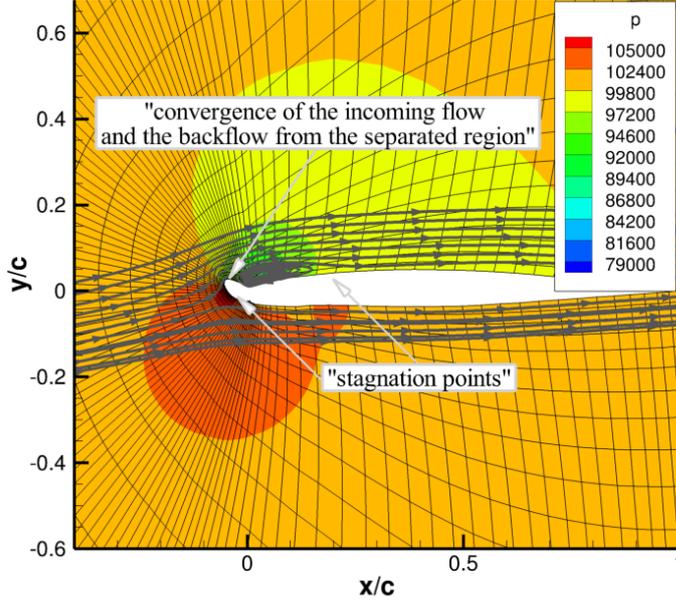
Fig. 2: *Flow around the predicted ice shape of case 11.*

structured grid further downstream). The level of refinement at the leading edge becomes about the same as for the structured grid for the fine case.

Table 3: *Unstructured grid description for case 9. $(\Delta x/c)_{LE}$ is the non-dimensional level of refinement on the wall at the leading-edge, where c is the chord length*

	Coarse	Medium	Fine	Very Fine	Ultra Fine
Overall number of cells	15000	29000	54000	89000	133000
Number of nodes on the wall	104	172	284	494	984
$(\Delta x/c)_{\text{LE}}$ (%)	1.5	0.75	0.375	0.1875	0.09375

Figures 4 and 5 show the mesh convergence for cases 9 and 13. A simple predictor approach was used for these simulations in order to remove the effect of remeshing and of the method of iced-surface displacement on the convergence. In these cases, the calculations with SIM2D and BLIM2D converge in a similar way. The convergence is rather satisfactory. It must be noticed that the results on the structured mesh are very close to the results on the "Very Fine" mesh. A slight

(a) *Coarse* **(b)** *Medium*

(c) *Fine* **(d)** *Very Fine*
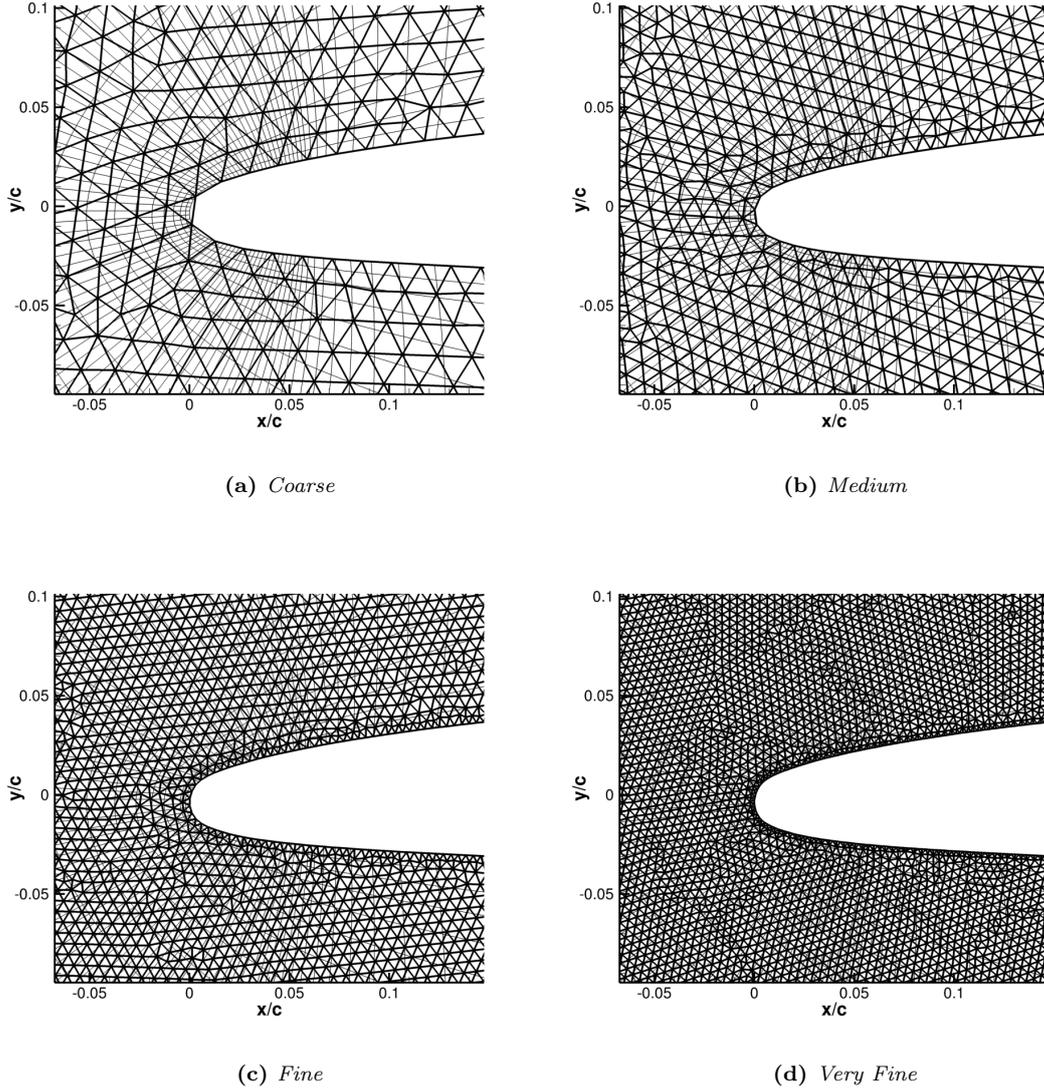
Fig. 3: *Unstructured grids used for the mesh convergence on case 9, predictor geometry.*

reduction of the ice thickness is observed for the "Ultra Fine" mesh, which is mainly due to the fact that the Smith and Spalding relationship (equation 4) triggers a singularity on the heat transfer (the integral term tends towards 0 at the stagnation point).

Regarding the effect of the time strategy (multi-step approach versus predictor-corrector approach), the convergence in the number of steps of the multi-step approach was assessed on case 9 on the 128-point structured mesh (also employed for the predictor-corrector approach, figure 6).This convergence is quite similar with BLIM2D and SIM2D: the result of the 50-step simulation is quite

(a) *SIM2D*   **(b)** *BLIM2D*
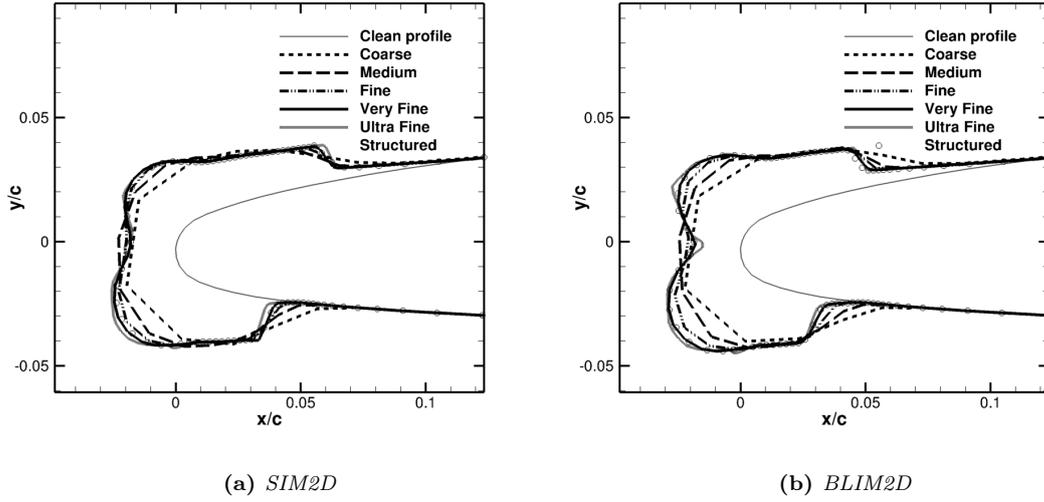
Fig. 4: *Mesh convergence: effect of the grid on the ice shape computed by IGLOO2D, using either SIM2D or BLIM2D, for case 9. Predictor computation.*

(a) *SIM2D*   **(b)** *BLIM2D*
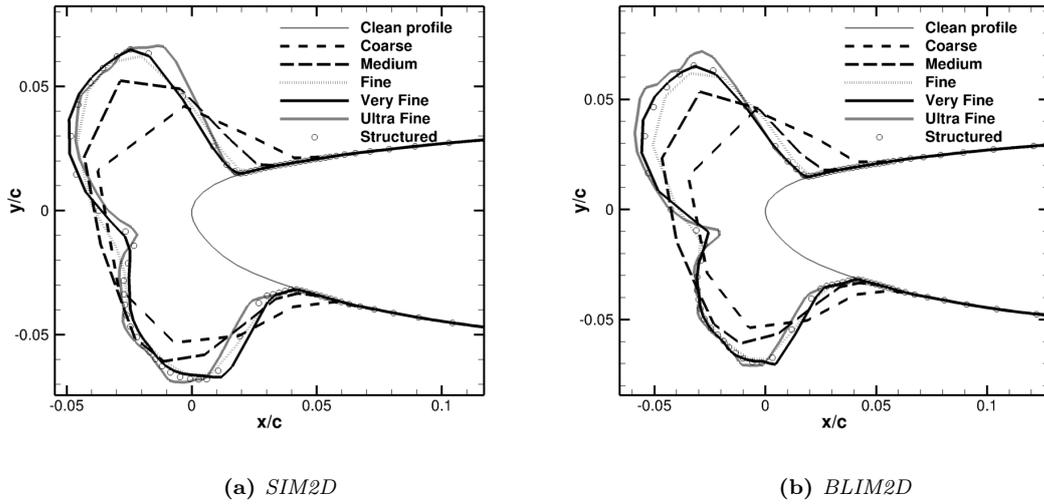
Fig. 5: *Mesh convergence: effect of the grid on the ice shape computed by IGLOO2D, using either SIM2D or BLIM2D, for case 13. Predictor computation.*

close to the one of the 80-step simulation (as also noted in article [3], where the calculations were made on a "Very Fine" unstructured grid), but not necessarily similar to the predictor-corrector results. The horn angles are here lower with the multi-step approach than with the predictor-corrector

17

method (the use of BLIM2D reinforces this observation with a slightly sharper decrease in the angle as the number of steps increases).

(a) *SIM2D*   (b) *BLIM2D*
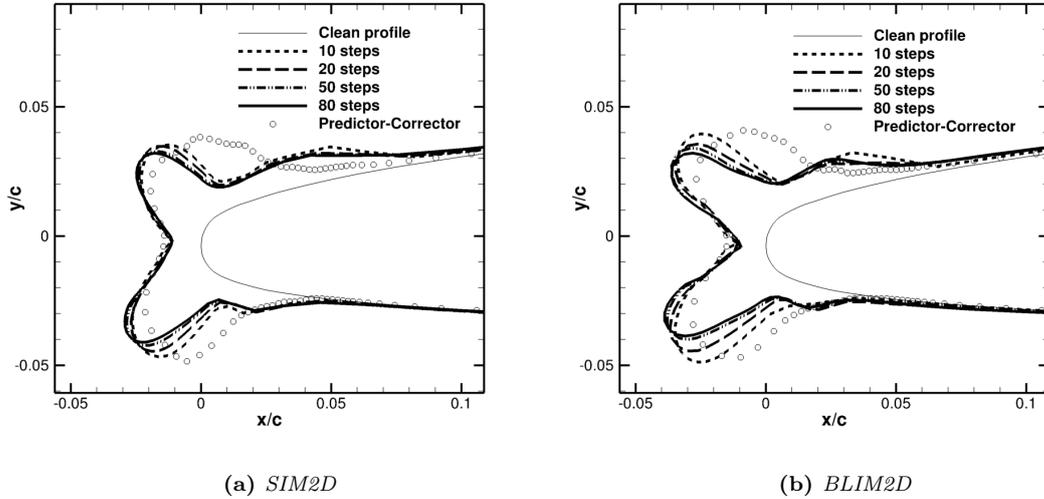
Fig. 6: *Convergence in the number of steps of the multi-step approach: effect of the number of steps on the ice shape computed by IGLOO2D, using either SIM2D or BLIM2D, for case 9.*

These results therefore show that, even if convergence is not perfect and the predictor-corrector and multi-step results do not necessarily match, the two codes compared in this article, BLIM2D and SIM2D, have very similar behaviors. In the following, the effect of the two codes will be compared using the predictor-corrector approach and the structured mesh (baseline IGLOO2D approach method based on empirical know-how that, from experience, has proven its ability to predict ice shapes).

C.   Effect on the heat transfer coefficient

Two kinds of IGLOO2D computations were performed: baseline IGLOO2D computations using SIM2D, and computations using BLIM2D instead of SIM2D.

The other calculation parameters were left identical, so that the impact of the boundary-layer-model change alone was evaluated. Consequently, as shown in figure 7 for conditions 1 (see table 1), the pressure coefficient C_p and the collection efficiency β are the same for both computations with BLIM2D and SIM2D ($s = 0$ at the stagnation point). This is actually only true for the predictor

step, where both computations are performed on the same grid (around the clean airfoil). For the corrector step (figure 8), there are some discrepancies because the grids around the predicted iced airfoil are not the same since different ice shapes are predicted.
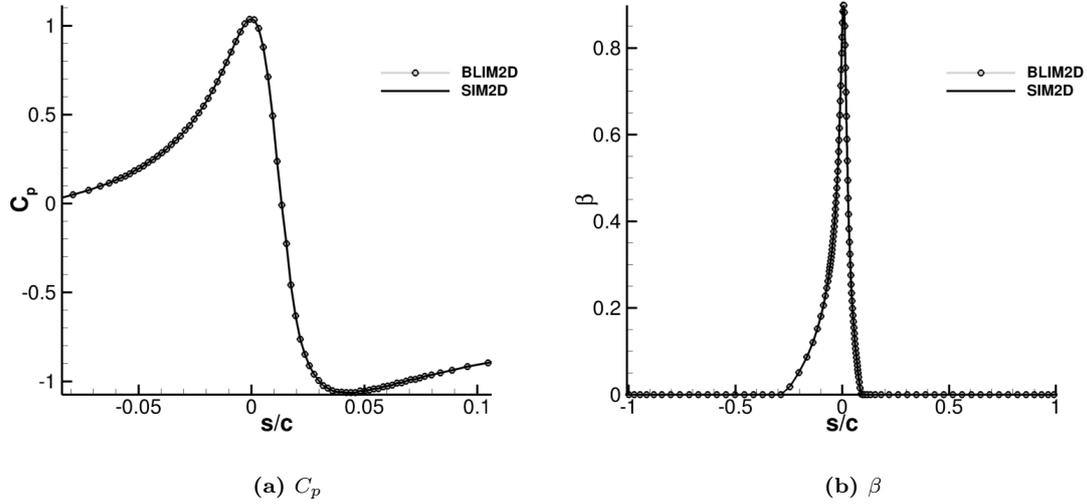

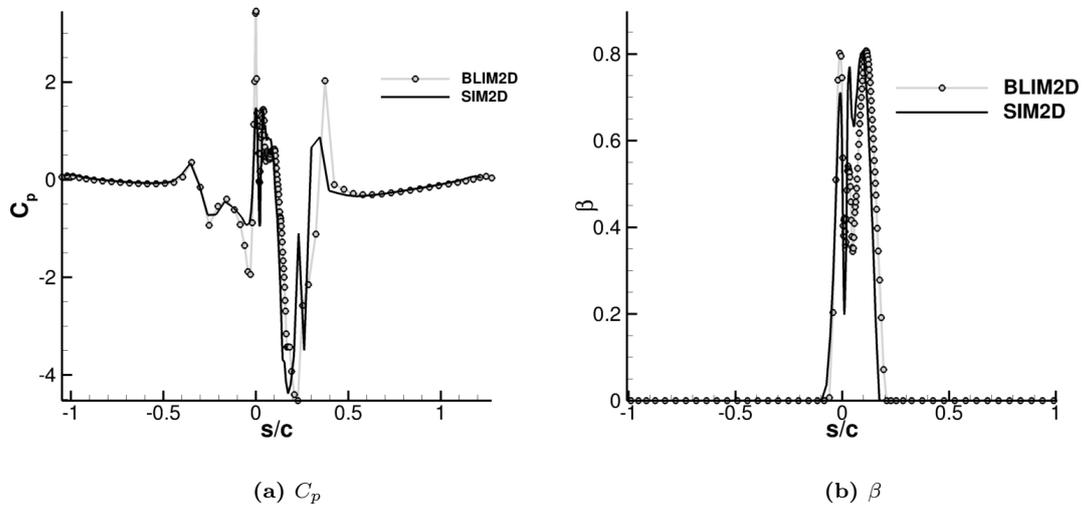**Fig. 7:** *Case 1: pressure coefficient C_p and collection efficiency β, predictor step.*

Fig. 8: *Case 1: pressure coefficient C_p and collection efficiency β, corrector step.*

The reason for the different ice shapes is the difference observed on the heat transfer coefficient h_{tc} (figure 9a). As shown in equation 5, h_{tc} is not the same with BLIM2D and SIM2D because

19

different momentum thicknesses θ are predicted by the two solvers (figure 9b).

(a) h_{tc} **(b)** θ
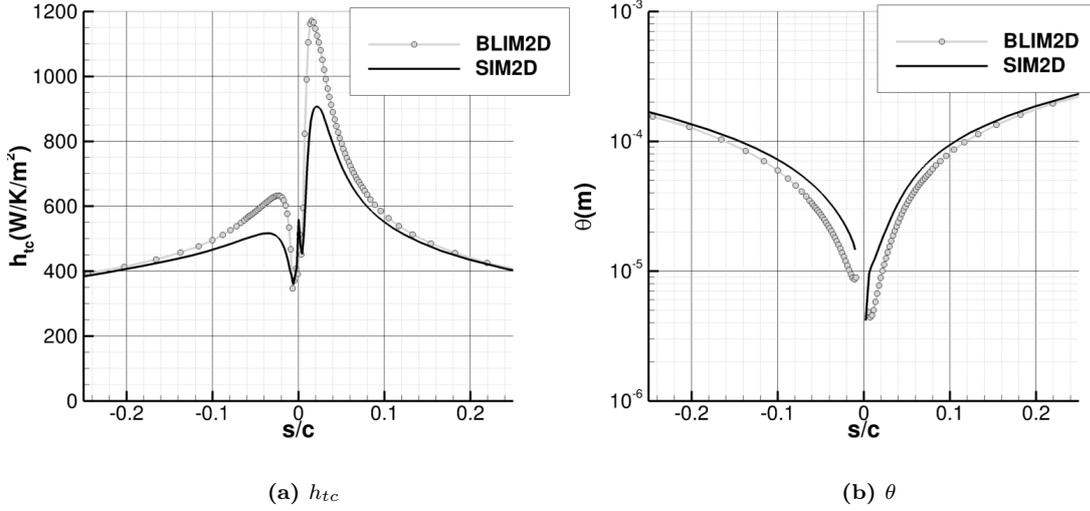
Fig. 9: *Case 1: heat transfer coefficient h_{tc} and momentum thickness θ, predictor step.*

By deriving equation 5 with respect to the variable θ, it is possible to show that the sensitivity of h_{tc} to θ is

$$\mathcal{S}_{h_{tc}} = \frac{\theta}{h_{tc}}\frac{\partial h_{tc}}{\partial \theta} = \alpha_{Cf}\mathcal{S}_{Cf} \tag{11}$$

where:

$$\alpha_{Cf} = \frac{\chi + 1/2}{\chi + 1}, \chi = \frac{\mathcal{P}r_{ts}St_k}{\sqrt{C_f/2}} \tag{12}$$

and:

$$\mathcal{S}_{Cf} = \frac{\theta}{C_f/2}\frac{\partial C_f/2}{\partial \theta} = -\frac{4214.6\dfrac{\theta}{k_s}\sqrt{C_f/2}}{\left(864\dfrac{\theta}{k_s} + 2.568\right)} \tag{13}$$

$\mathcal{S}_{h_{tc}}$ is a function of the two parameters k_s^+ and θ/k_s. Equations 11, 12 and 13 and figure 10 show that $\mathcal{S}_{h_{tc}}$ is negative. A decrease in θ thus leads to increasing h_{tc}. The sensitivity is largest for intermediate values of θ/k_s. The range of k_s^+ and θ/k_s observed in the IGLOO2D computations of the two databases (tables 1 and 2) are also shown in figure 10. To be more specific, the values observed in the "predictor" step of the BLIM2D computations in the iced region and in turbulent

regime are shown in figure 10. The point cloud is in the range of k_s^+ from 30 to 1000 and for θ/k_s in the range from 0.001 to 1. For this range of data,the sensitivity is shown to be higher (in absolute value) for lower values of θ/k_s (and lower values of k_s^+). Notice that for a given run, k_s^+ and θ/k_s generally increase simultaneously from the stagnation point to the trailing edge. It is thus in the vicinity of the leading edge that the sensitivity is higher.
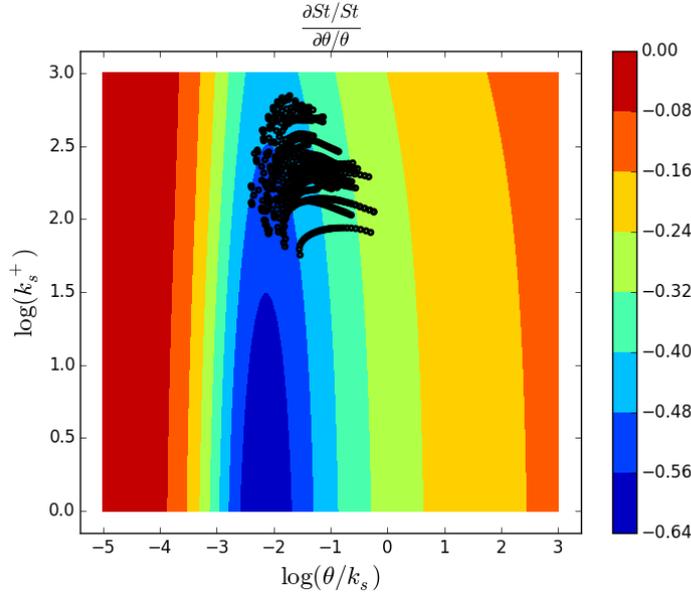

Fig. 10: *Sensitivity of h_{tc} to θ. Scatter: computational values observed in the IGLOO2D calculations of databases 1 and 2.*

In many cases, the relative change in θ is found to be large. Consequently, it is logical that a large relative change is also observed on h_{tc}. In the worst case investigated (case 1), the use of BLIM2D results in a 60 % reduction in θ and a 45 % increase in h_{tc} is logically observed.

Figures 11, 12, 13 and 14 confirm that θ and h_{tc} are affected in all cases, here cases 2, 10, 11 and A. For all cases, θ is predicted smaller by BLIM2D than by SIM2D (negative relative change). This is logical because the effect of the pressure gradient is neglected by SIM2D: the pressure gradient is favorable near the leading edge (the edge-velocity gradient is positive), which tends to decrease θ according to the von Kármán equation (first line of equation 10 at steady-state). BLIM2D thus produces larger heat transfer coefficients than SIM2D. Maximum increase is obtained at the beginning of the turbulent region (up to around 40 % relative difference, for peak relative change

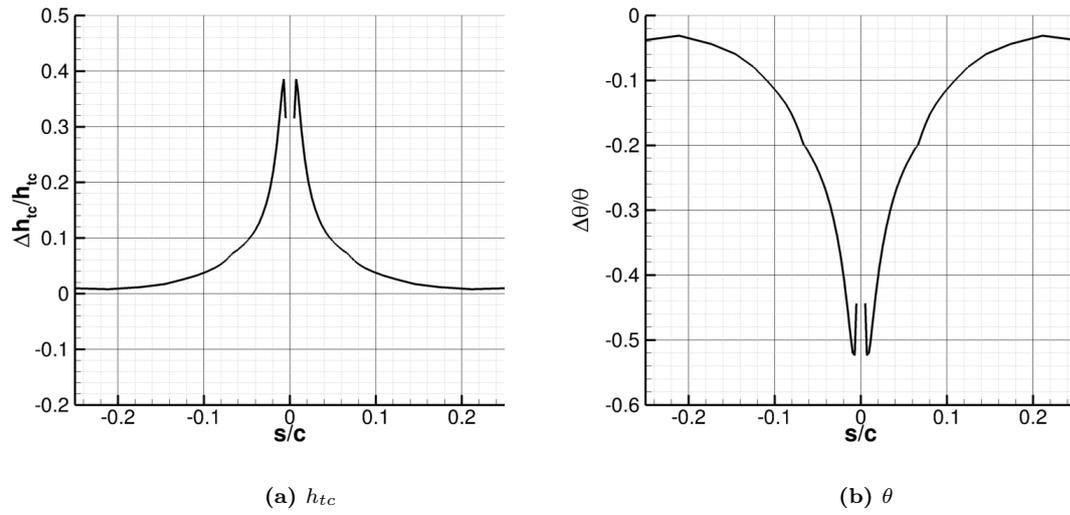21

in θ around 60 %), which is consistent with figure 10.

(a) h_{tc} **(b)** θ

Fig. 11: *Case 2: relative change in heat transfer coefficient h_{tc} and momentum thickness θ in turbulent region, predictor step.*
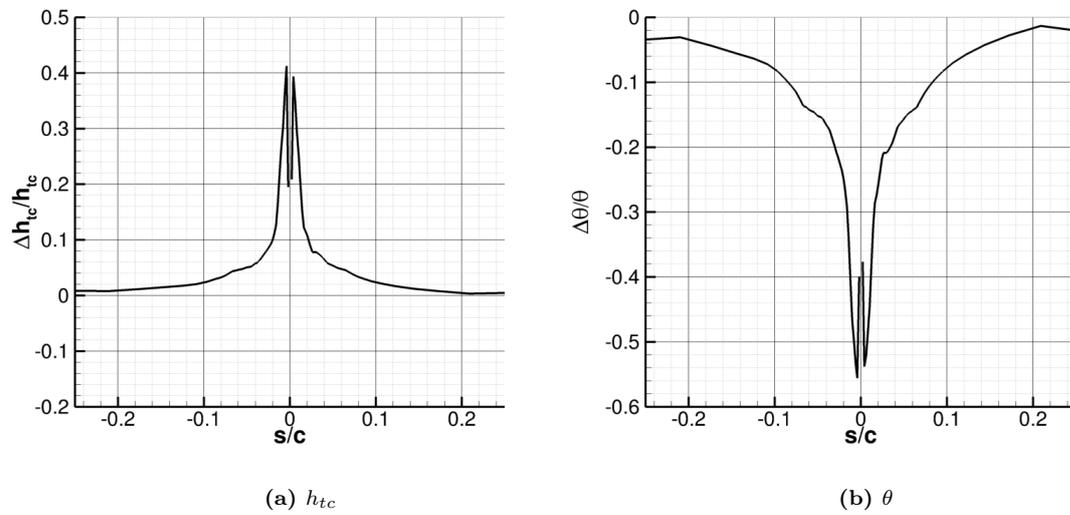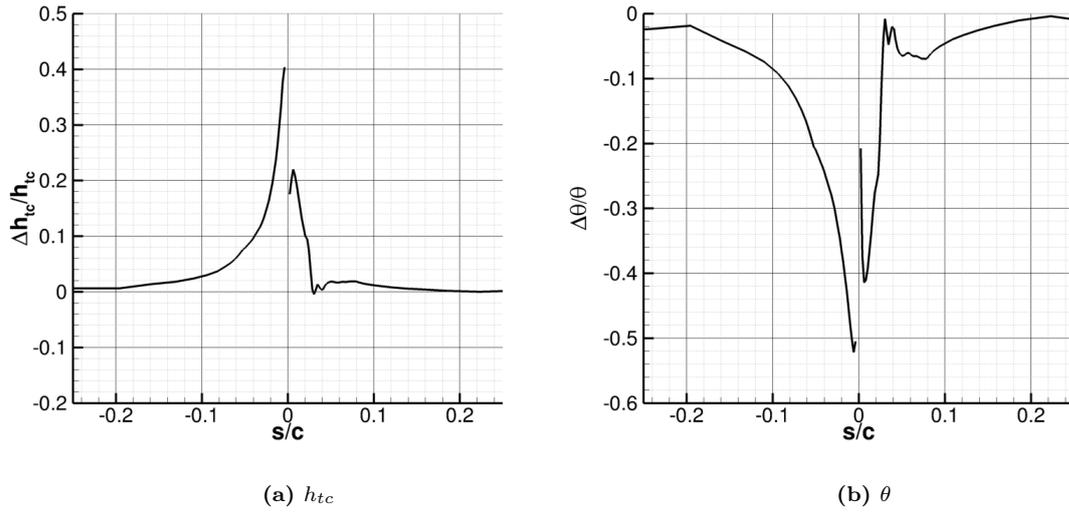

(a) h_{tc} **(b)** θ

Fig. 12: *Case 10: relative change in heat transfer coefficient h_{tc} and momentum thickness θ in turbulent region, predictor step.*

(a) h_{tc} (b) θ

Fig. 13: *Case 11: relative change in heat transfer coefficient h_{tc} and momentum thickness θ in turbulent region, predictor step.*
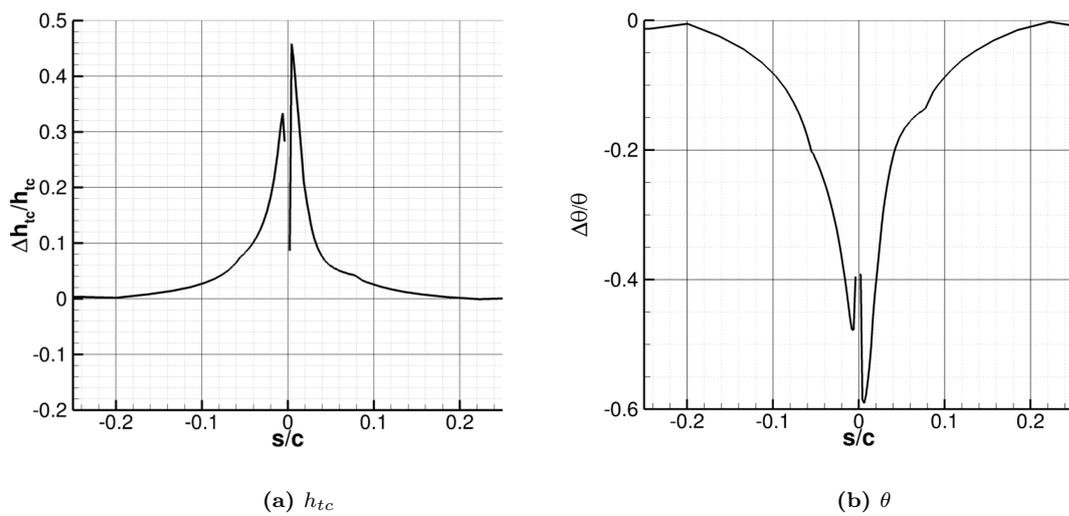

(a) h_{tc} (b) θ

Fig. 14: *Case A: relative change in heat transfer coefficient h_{tc} and momentum thickness θ in turbulent region, predictor step.*

D. Effect on the ice shapes

1. Method

Both qualitative and quantitative assessments of the impact on the ice shapes of the use of BLIM2D instead of SIM2D are performed in this section. Figures 19, 20, 21, 25, 26, 27 and 28 show the ice shapes produced by IGLOO2D and the experimental ice shapes. Sometimes, there are several experimental ice shapes for one icing condition, which allows to observe some repeatability errors. This kind of picture allows to qualitatively assess the effectiveness of the codes. The horn angles α_l and α_u are also shown in the figures. They are defined as shown in figure 15. The horns are here defined as the locations where the ice thickness is maximum on both lower side and upper side. It should be noted that the horn angles were assessed from the cell-centered ice growth rate, which is the primary variable of the Messinger balance. By contrast, the ice shapes were plotted by displacing the nodes of the surface grid (and not by cell-centroid displacement). Consequently, there can be small deviations between the horn angles and the ice shapes, as shown for instance in figure 20a.
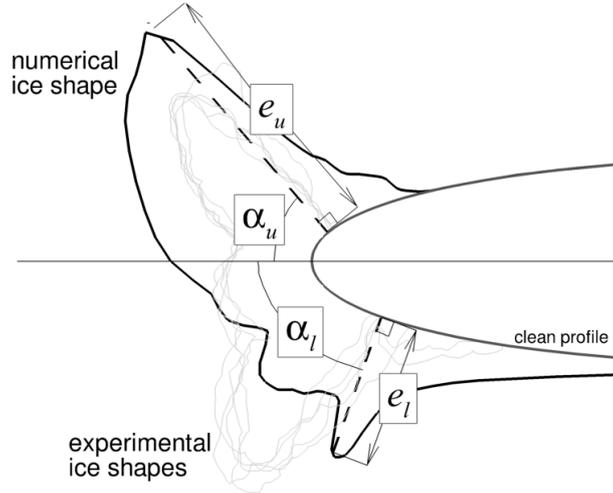
Fig. 15: *Ice shape analysis: definitions. Ice shape of case 1.*

As in articles [3] and [25] for instance, a quantitative assessment is therefore also made. The horn

angles and heights (e_l and e_u) produced by BLIM2D and SIM2D are compared against experimental values, as well as the ice cross-section area (which represents the ice volume). Figures 16, 17, 18, 22 and 23 thus show the relative errors between the data predicted by simulations ϕ_{num} and those measured experimentally ϕ_{exp}, $\dfrac{|\phi_{\mathrm{num}} - \phi_{\mathrm{exp}}|}{|\phi_{\mathrm{exp}}|}$, where ϕ is either the horn angle α, the horn height e or the ice section area.

2. First database

Figure 16a shows that for most cases, BLIM2D produces lower relative error in the upper side horn angle than SIM2D (compared to the experiment). Better results are also given overall by BLIM2D for the lower side horn angle (figure 17a). Regarding the horn sizes, BLIM2D can also be deemed as slightly better than SIM2D (especially on lower side, figure 17b). However, regarding the ice area (figure 18), the results of BLIM2D and SIM2D are often very similar but SIM2D is slightly better than BLIM2D. The article [3] already mentioned that IGLOO2D tends to overpredict the ice volume. It was indeed suspected that not all the water could be conserved on the ice surface, especially on the horns. The absence of a water shedding model was therefore blamed for that. Thus, the boundary layer resolution should not be involved in this issue.
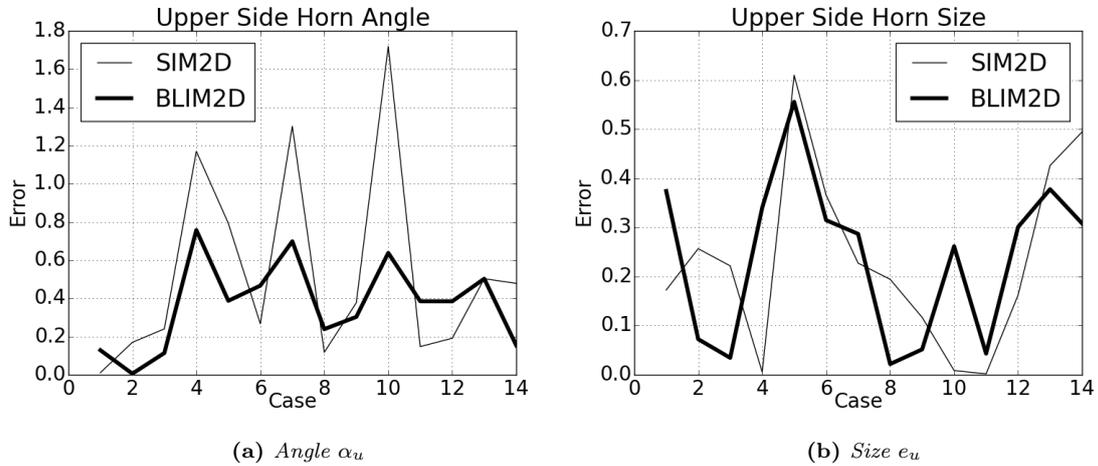

(a) Angle α_u (b) Size e_u

Fig. 16: *Cases 1 to 14: comparison between BLIM2D and SIM2D for upper side horn.*

After figures 16 and 17, the cases for which BLIM2D is not clearly better than SIM2D are cases 1, 10, 11, 12 and 14. Figure 19 shows that for cases 1 and 12, neither BLIM2D nor SIM2D have a
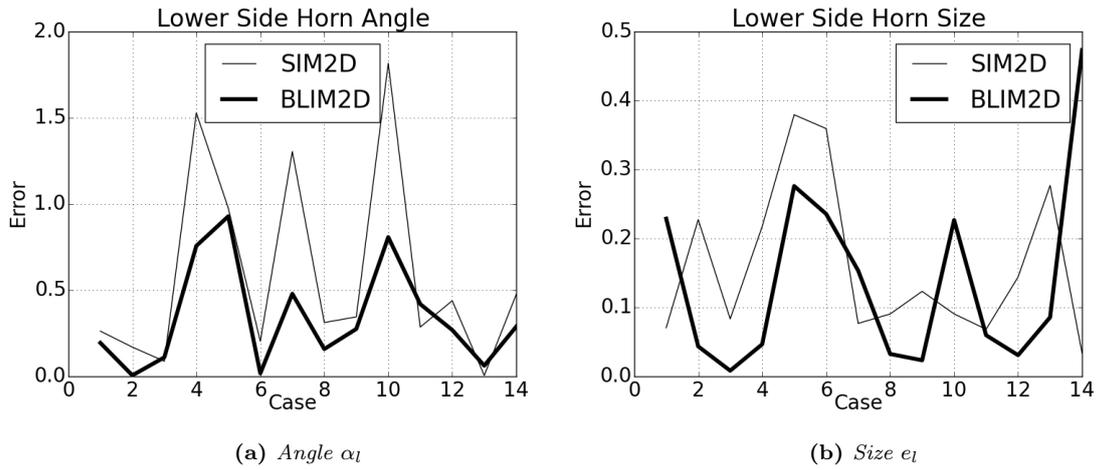
(a) *Angle α_l*

(b) *Size e_l*

Fig. 17: *Cases 1 to 14: comparison between BLIM2D and SIM2D for lower side horn.*
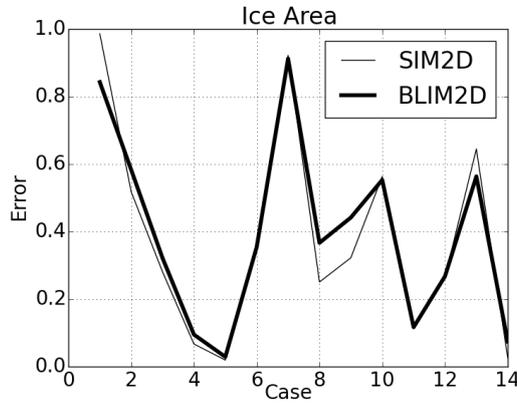

Fig. 18: *Cases 1 to 14: comparison between BLIM2D and SIM2D on predicted ice section area.*

clearly better prediction than each other. Regarding cases 10 and 14 (figure 20), BLIM2D predicts better horn angles than SIM2D but the horn thickness is not correct (note, however, that there is a low level of experimental repeatability for case 14). For case 11 only, SIM2D may be judged as slightly better than BLIM2D (especially because of the upper-side horn, figure 21).
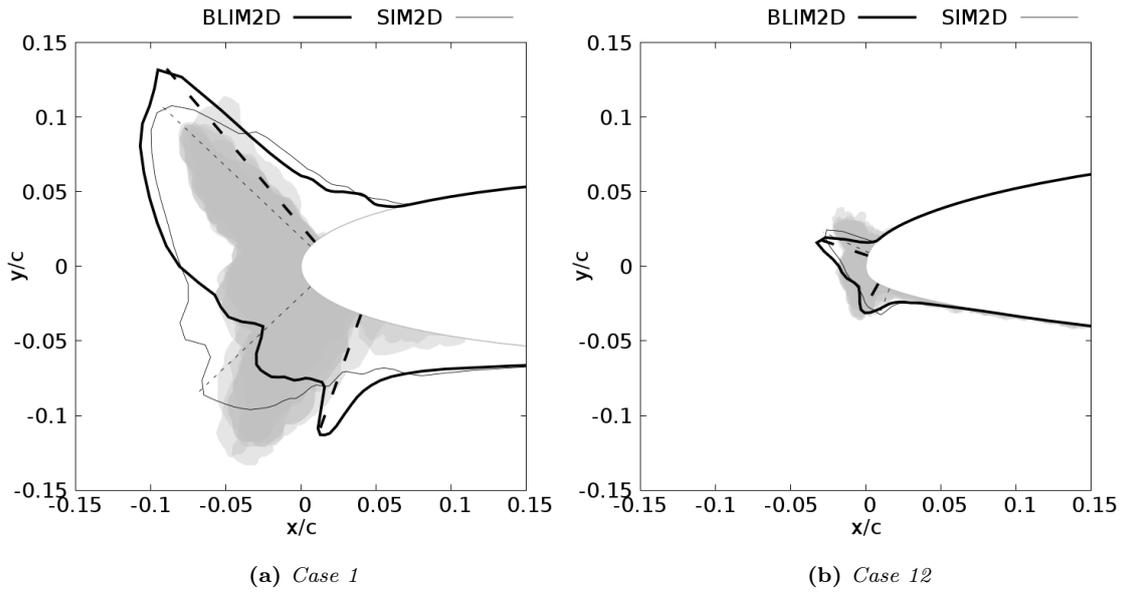
(a) *Case 1*

(b) *Case 12*

Fig. 19: *Cases 1 and 12: ice shapes produced by BLIM2D and SIM2D.*

(a) *Case 10*
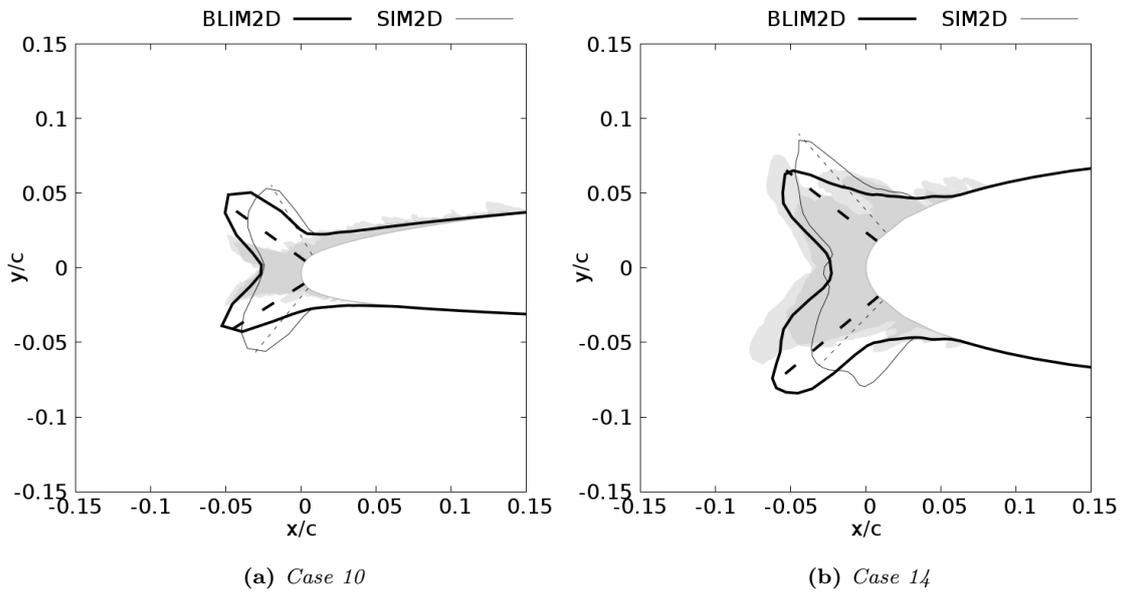
(b) *Case 14*

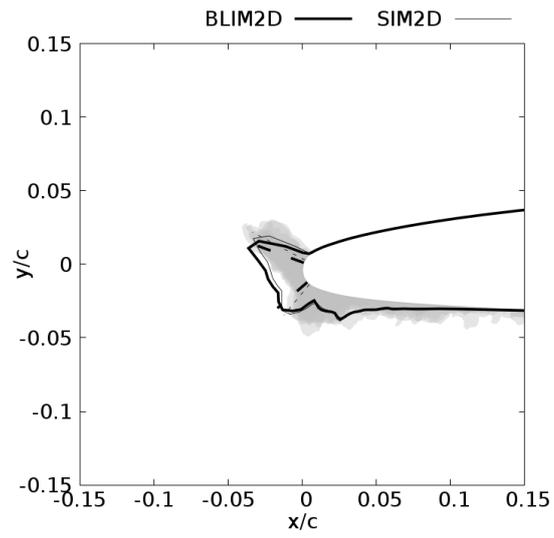**Fig. 20:** *Cases 10 and 14: ice shapes produced by BLIM2D and SIM2D.*

Fig. 21: *Case 11: ice shapes produced by BLIM2D and SIM2D.*

On the second database (Table 2), BLIM2D did not improve the ice shapes as much as on the cases of the first database. But it should be recalled that the ice shapes produced by IGLOO2D using SIM2D were already quite good for the test-cases of the second database. Figure 22 shows that the use of BLIM2D leads to better-predicted upper-side-horn angles. But the lower-side-horn angles seem to be better predicted with SIM2D (figure 23). The errors on the ice section area (figure 24) are generally lower than on the first database, which may be due to the fact that less water may be shed at the horn tips for this simpler database. In addition, BLIM2D is generally slightly better than SIM2D on the ice section area criterion (especially in cases D and E where too much ice volume is expected by SIM2D).
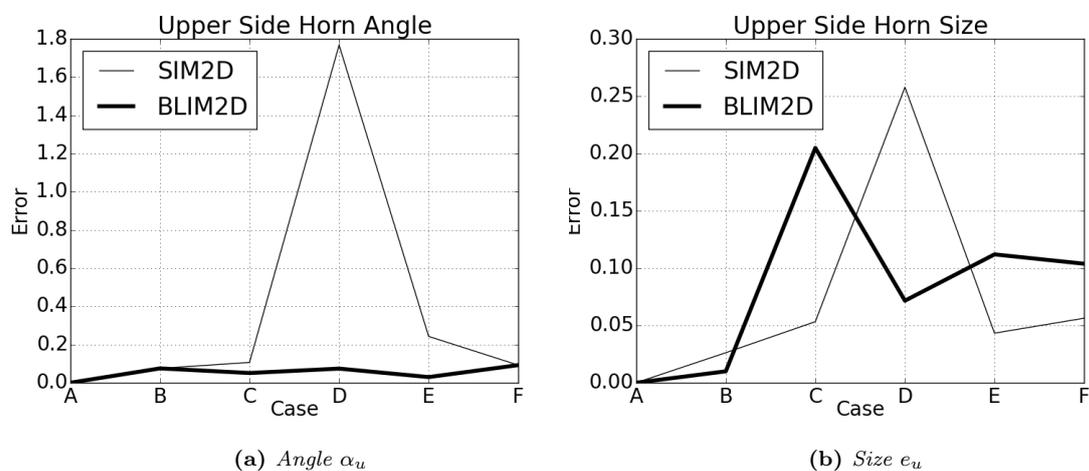

(a) Angle α_u (b) Size e_u

Fig. 22: *Cases A to F: comparison between BLIM2D and SIM2D for upper side horn.*

The cases for which BLIM2D does not improve the ice shapes after figures 22 and 23 are test-cases C to F. Cases C and D look clearly better predicted with SIM2D (figure 25). Regarding the upper-side horn angle of case D, the high relative error shown in figure 22a actually comes from the low expected angle. The two ice shapes predicted with BLIM2D and SIM2D look pretty similar on case F (figure 26b). Finally, for case E, the upper horn is better predicted with BLIM2D while the lower horn is better predicted with SIM2D (figure 26a).
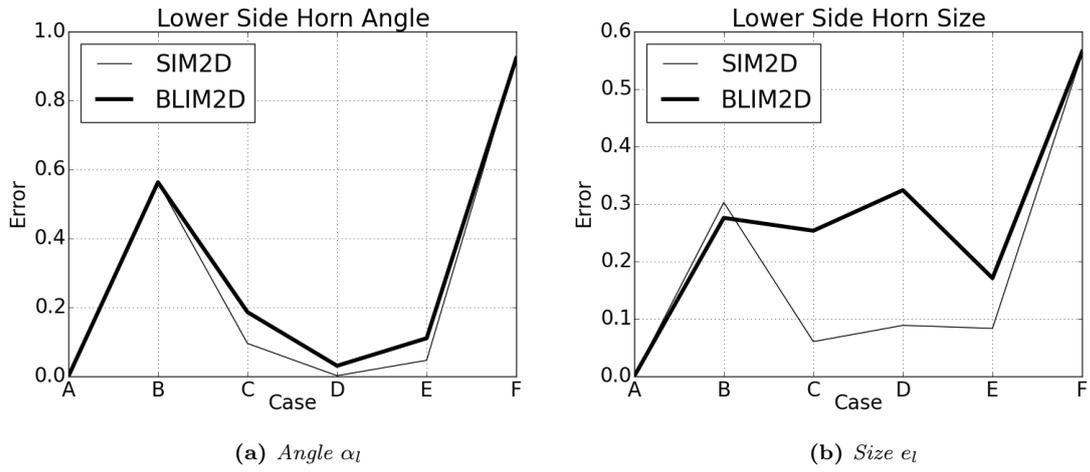
(a) *Angle α_l*

(b) *Size e_l*

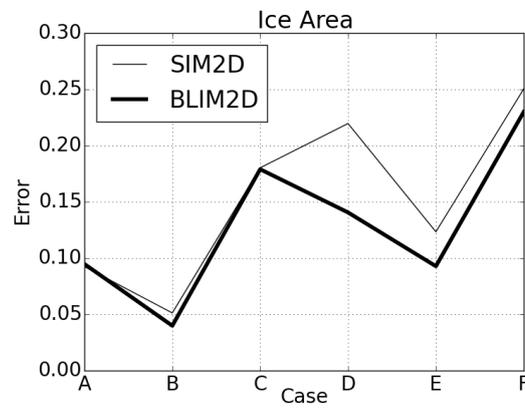**Fig. 23:** *Cases A to F: comparison between BLIM2D and SIM2D for lower side horn.*

Fig. 24: *Cases A to F: comparison between BLIM2D and SIM2D on predicted ice section area.*
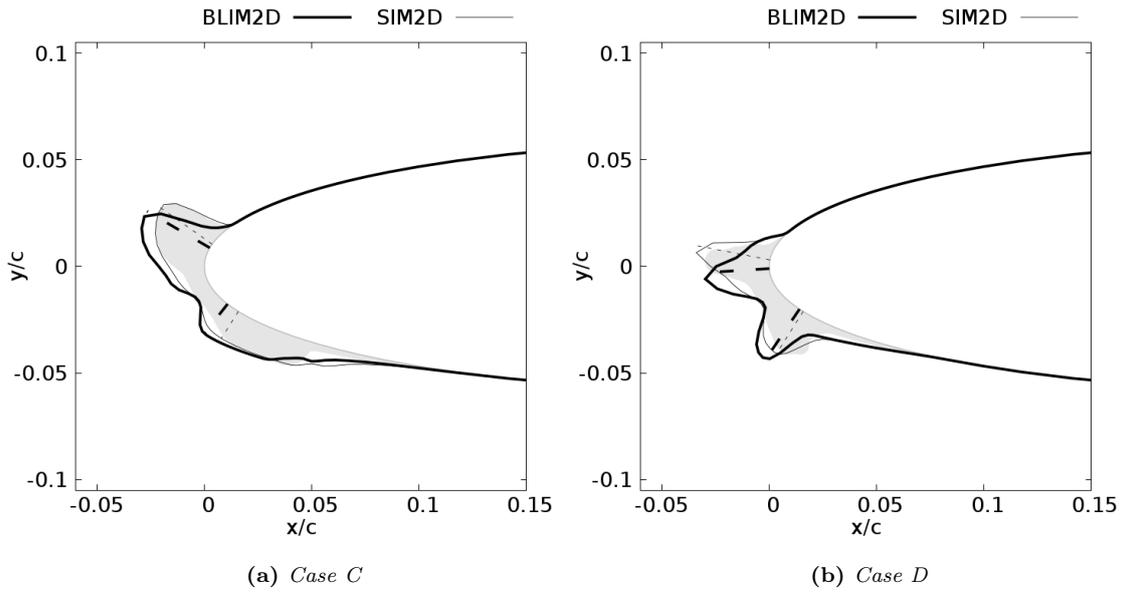
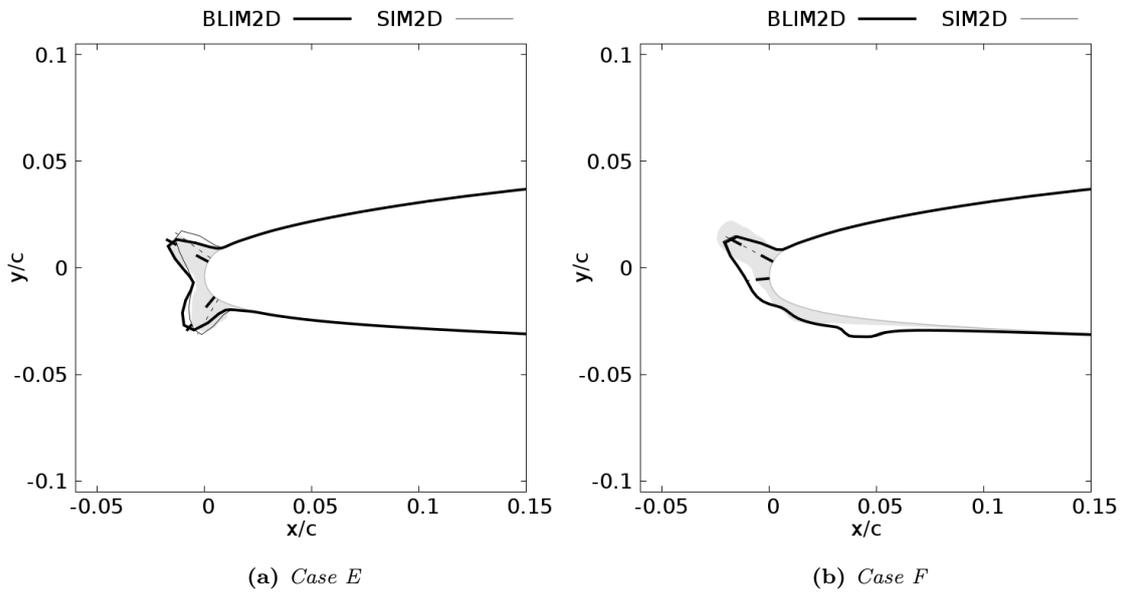**Fig. 25:** *Cases C and D: ice shapes produced by BLIM2D and SIM2D.*

(a) *Case C*

(b) *Case D*

(a) *Case E*

(b) *Case F*

Fig. 26: *Cases E and F: ice shapes produced by BLIM2D and SIM2D.*

E. Discussion

BLIM2D includes more physics than SIM2D, especially the influence of the pressure gradient. In particular, it is necessary to assume a constant value for the shape factor $H = \dfrac{\delta_1}{\theta} = 1.29$ and a simple power law $u/u_e(y) = (y/\delta)^{1/n}$ where $n = 7$ to derive equation 1 (used in SIM2D) from the von Kármán equation. These values are often used in the literature and quite representative of zero-pressure-gradient flows, although H and n usually vary along a zero-pressure-gradient flat plate. Additionally, an approximate law was also employed for the skin friction coefficient to derive equation 1. The computations on the two databases show that BLIM2D generally improves the results, sometimes significantly compared to SIM2D (for example, see figures 27 and 28). Moreover, BLIM2D never substantially worsens the prediction of ice shapes. A significant improvement often happens when SIM2D predicts large horn angles whereas BLIM2D reduces the horn angles. This is due to the fact that BLIM2D generally predicts lower θ and larger h_{tc} than SIM2D (and thus stronger cooling due to convection). In general, the prediction of the upper horn angle is also improved by BLIM2D.

However, at this stage, given that the momentum thickness is not the only physical parameter influencing the heat transfer coefficient, it is difficult to draw a general conclusion on the conditions under which BLIM2D will systematically provide better results than SIM2D.

It should also be noted that the method does not significantly increase the computational time compared to the simplified method. Table 4 shows a fairly representative estimate of the computational times per solver for case 1. BLIM2D certainly increases the computational time compared to SIM2D, but in a way that does not penalize the whole simulation. Most of the computational time is indeed due to the EULER2D and TRAJL2D simulations. It should also be mentioned that BLIM2D has not been optimized (no convergence acceleration technique was implemented for example). Complex cases thus lead to an increase in computational time (example of the corrector calculation 15 times more costly than the predictor calculation in table 4) because the CFL number must be reduced in the areas where the problem becomes steep. It must also be recalled that the general method is less time-consuming than that of using a Navier-Stokes calculation (it is also easier to generate the grid for inviscid calculations than for Navier-Stokes

32

simulations).

Table 4: *Percentage of computational time spent in each solver, case 1.*

Computation	STRMESH2D	EULER2D	BLIM2D/SIM2D	TRAJL2D	MESSINGER2D
SIM2D, predictor	3.5	47.7	0.1	48.2	0.4
SIM2D, corrector	1.5	55.2	0.2	41.4	1.6
BLIM2D, predictor	0.7	49.0	0.9	49.0	0.4
BLIM2D, corrector	1.4	49.9	15.3	32.1	1.4

The improvement given by BLIM2D is one step among several tasks aimed at improving the heat transfer modelling over iced surfaces. Improving the modelling of roughness height distribution [23, 24] is another key issue that will have to be addressed in IGLOO2D. This may lead to a reduction of k_s in the vicinity of the leading edge (thus probably reducing h_{tc}) and a change in the rougher area that cannot be quantified for sure so far. Another consequence will be the probable shifting of the laminar-turbulent transition, to be combined with the presence of a large transitional area [18], thus again probably reducing h_{tc}.

The use of more evolved or more recent rough-wall models for h_{tc} than the Makkonnen model is also expected. For instance, the article [22] described an adaptation of Aupoix's model to the framework of integral boundary layer methods. One important conclusion of the latter work was that the rough-wall momentum thickness should be used as an input of the new h_{tc} model (so-called integral Aupoix-Grigson-Colebrook model). The framework of the full integral boundary layer model exposed in section III is much more general than the simplified integral model shown in section II B. It should therefore be much easier to implement a rough-wall closure model and use the integral Aupoix-Grigson-Colebrook model in BLIM2D than in SIM2D.

The effect of the time strategy should also be further investigated. Section IV B showed that the choice of either the predictor-corrector or the multi-step approach affects the prediction of the ice shape. It must be noted that the multi-step method did not systematically produce better results in comparison to the experimental ice shapes. But the way BLIM2D affects the ice shapes compared

to SIM2D with this approach needs to be studied in more detail.
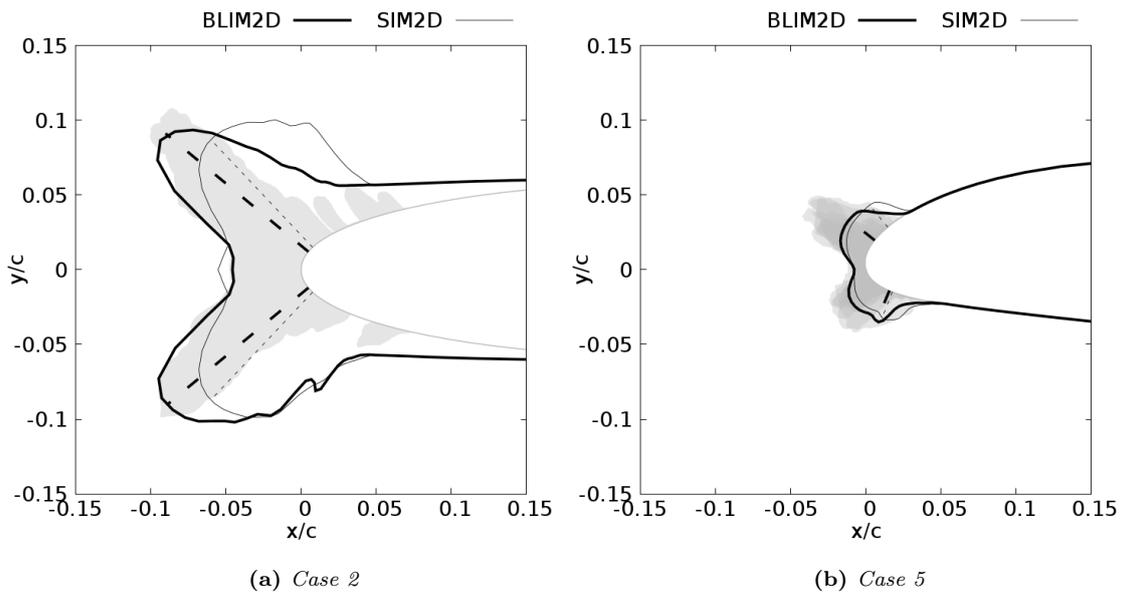

(a) *Case 2*

(b) *Case 5*

Fig. 27: *Cases 2 and 5: ice shapes produced by BLIM2D and SIM2D.*
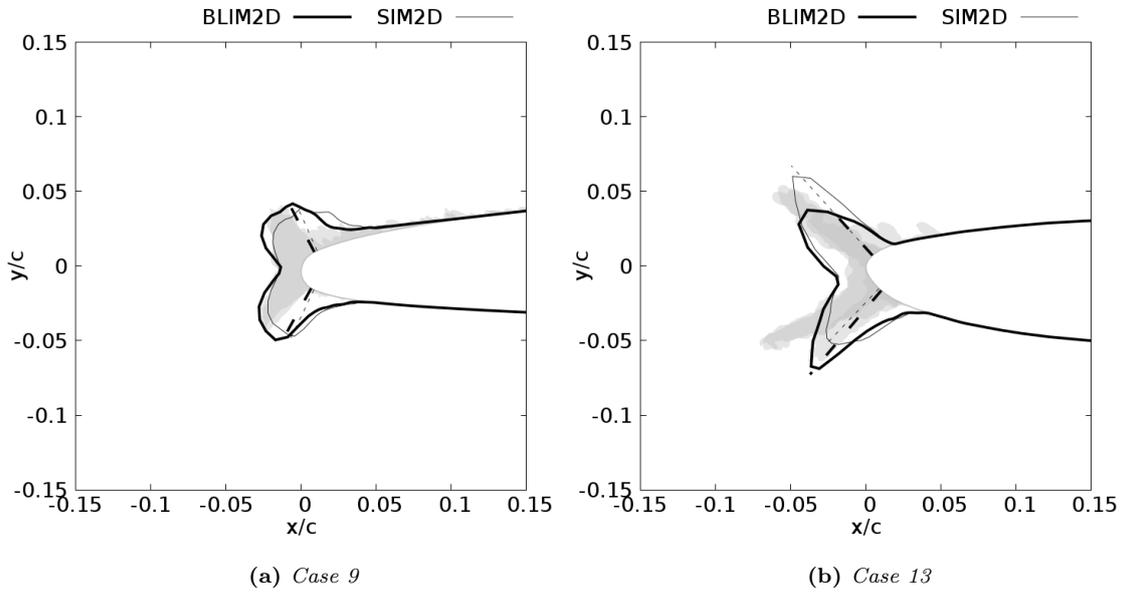

(a) *Case 9*

(b) *Case 13*

Fig. 28: *Cases 9 and 13: ice shapes produced by BLIM2D and SIM2D.*

V. Conclusion

The ONERA's icing suite IGLOO2D was used to compute 20 ice shapes, by varying the boundary layer calculation method. The widely-used simplified integral method was used as a reference. On the other hand, the more general integral boundary layer method developed in [6] was used. The latter approach is expected to be more accurate, more flexible and expandable in 3D, in rough-wall conditions or in non-uniform wall temperature conditions for instance.

The present article shows that the new method generally produces lower smooth-wall momentum thicknesses than the simplified method. The heat transfer coefficient (computed using the Makkonen model) is thus generally predicted larger by the new method. The ice shapes therefore often exhibit lower horn angles, since the ice cooling by the airflow is larger. For the test-cases investigated in this article, the ice shapes are often better predicted with the new method. So far, the use of BLIM2D has mainly allowed to improve the accuracy of the momentum-thickness calculation. However, this parameter is not the only one to have an influence on the heat exchange coefficient and in turn on the ice accretion rate. There are therefore cases in which the use of BLIM2D did not bring any real progress in the prediction of ice shapes. Further improvements will be required by studying more databases and working on other aspects of ice-accretion modelling.

The method presented in the present article should thus be seen as a first step in a set of tasks to better represent icing phenomena. Progress is now expected on other aspects to improve the ice shape prediction. First, still in connection to the boundary layer modelling, the modelling of roughness size (and distribution) as well as laminar-turbulent transition and heat transfer on rough surfaces must be improved. Second, complex runback (rivulets, beads) has to be modelled carefully, since it changes the surface wetness and the overall convective exchange with the airflow. Finally, as far as numerical aspects are concerned, the unsteady growth of the ice shape (multi-shot computations with controlled time discretization and ice layer transport) is still a challenging issue, especially regarding the extension to 3D.

APPENDIX A: BRIEF DESCRIPTION OF THE MAIN MODULES OF THE IGLOO2D ICING SUITE

1. MESSINGER2D

MESSINGER2D classically solves the so-called Messinger balance equations on the mass and energy of water on the icing surface. The balance accounts for impinging water, runback, evaporation, freezing and convection due to the airflow. For the simulations of this article, the wall was assumed adiabatic: no ice protection systems were modelled and the heat exchange between the ice and the wall was neglected. The two equations are solved iteratively from the stagnation point to the trailing edge, in order to produce the wall temperature distribution, the ice growth rates and the runback mass rate. Since there are two equations and three unknowns, a third equation is used, depending on the icing regime. Four regimes can be obtained (rime ice: there is no liquid water and the runback mass rate is therefore 0; glaze ice: there are both liquid and solid water and the temperature is therefore 273.15K; running-wet regime: there is no solid water and the ice growth rate is therefore 0; dry-wall: there is no water). The reader will find further information about the specific implementation done in MESSINGER2D in reference [3].

2. TRAJL2D

TRAJL2D is the Lagrangian module of IGLOO2D [3] to solve the trajectories of the water droplets, under the assumption of disperse phase (which is obviously met because the droplet volume fractions are around 10^{-6}). The volume fraction is so small that a 1-way approach is used: the airflow drives the droplets and the retroaction of droplets on the airflow is neglected. The Schiller-Naumann correlation is used for the drag coefficient of droplets (considered as perfectly spherical). Like most Lagrangian solvers, numerical particles are tracked rather than all of the droplets in the cloud. 10000 numerical particles were used for this article, which is sufficient to quickly obtain a converged solution because the injection is done along a restricted emission line. An algorithm is indeed used to restrict the emission line to an area closely surrounding the region from which the emitted droplets impinge on the iced surface. Since a steady solution is expected, a local-time-stepping procedure is used and a first-order time-resolution is performed. Finally, only clouds of small droplets, characteristic of the so-called Appendix C of the icing certification rules, are addressed in this article. But some

supercooled-large-droplet and ice-crystal models are available in TRAJL2D [3].

3. STRMESH2D

As further described in reference [3], there are two structured mesh generation methods in IGLOO2D: the isoparametric coordinate mapping [27] and the resolution of a hyperbolic partial differential equation [28]. The standard approach used for this article was the first method. This method consists of deriving a unique mapping between the Cartesian coordinates of the parametric space and the curvilinear coordinates of the real space. It is rather efficient but it sometimes fails for ice shapes with large curvatures. For the simulations of this article, when the first method failed, the hyperbolic method was used. This method consists of solving a partial differential equation to generate a grid from the iced surface by marching in the normal direction. For both kinds of grids, there are 128 cells on the iced surface and 4200 quadrilateral elements overall. The grid is extended about 6 chord lengths around the airfoil. The mesh is refined in the vicinity of the leading edge (the cell size is around 0.002 c in the first 5% of the airfoil, where c is the chord length) and coarsened in the vicinity of the trailing edge (around 0.02 c).

REFERENCES

[1] Wright, W. B., "User's manual for LEWICE version 3.2," NASA/CR-2008-214255.

[2] Guffond, D. and Brunet, L., "Validation du programme bidimensionnel de captation," Tech. rep., Office National D'Etudes et de Recherches Aerospatiales, Chatillon Cedex, France, RP 20/5146 SY, 1988.

[3] Trontin, P., Kontogiannis, A., Blanchard, G., and Villedieu, P., "Description and assessment of the new ONERA 2D icing suite IGLOO2D," 9th AIAA Atmospheric and Space Environments Conference - AVIATION 2017 DENVER, USA, AIAA 2017-3417, 2017, doi:10.2514/6.2017-3417.

[4] Gent, R. W., Dart, N. P., and Cansdale, J. T., "Aircraft icing," *Philosophical Transactions of the Royal Society of London*, Vol. 358, 2000, pp. 2873–2911, doi:10.1098/rsta.2000.0689.

[5] Thwaites, B., "Approximate calculation of the laminar boundray layer," *Aero. Quarterly*, Vol. 1, No. 3, 1949, pp. 245–280, doi:10.1017/S0001925900000184.

[6] Bayeux, C., Radenac, E., and Villedieu, P., "Theory and Validation of a 2D Finite-Volume Integral Boundary-Layer Method for Icing Applications." *AIAA Journal*, Vol. 57, No. 3, 2019, pp. 1092–1112, doi:10.2514/1.J057461.

[7] Tai, T. C., "An Integral Prediction Method for Three-Dimensional Flow Separation." 22nd AIAA Aerospace Sciences Meeting, AIAA 84-0014, 1984, doi:10.2514/6.1984-14.

[8] Mughal, B., *Integral Method for Three-Dimensional Boundary-Layers*, Ph.D. thesis, Massachussetts Institute of Technology, 1998.

[9] Nishida, B. A., *Fully simultaneous coupling of the full potential equation and the integral boundary layer equations in three dimensions*, Ph.D. thesis, Massachusetts Institute of Technology, 1996.

[10] Milewski, W. M., *Three-dimensional viscous flow computations using the integral boundary layer equations simultaneously coupled with a low order panel method*, Ph.D. thesis, Massachusetts Institute of Technology, 1997.

[11] van Es, B., "Comparison and Application of Unsteady Integral Boundary Layer Methods using various numerical schemes," Tech. rep., Faculty of Aerospace Engineering, Delft University of Technology, 2009.

[12] DiPaola, M. and Willis, D., "A rotating reference frame, integral boundary layer method," 46th AIAA Fluid Dynamics Conference, AIAA 2016-3974, 2016, doi:10.2514/6.2016-3974.

[13] Lokatt, M. and Eller, D., "Robust viscous-inviscid interaction scheme for application on unstructured meshes." *Computers and Fluids*, Vol. 145, No. 2, 2017, pp. 37–51, doi:10.1016/j.compfluid.2016.12.012.

[14] Zhang, S., Galbraith, M., Allmaras, S., Drela, M., and Darmofal, D. L., "A Non-parametric Discontinuous Galerkin Formulation of the Integral Boundary Layer Equations with Strong Viscous/Inviscid Coupling." 23rd AIAA Computational Fluid Dynamics Conference - AVIATION 2017 DENVER, USA, AIAA 2017-4278, 2017, doi:10.2514/6.2017-4278.

[15] Beaugendre, H., *A PDE-Based 3D Approach to In-Flight Ice Accretion*, Ph.D. thesis, McGill University, 2003.

[16] Hasanzadeh, K., Mosahebi, A., Laurendeau, E., and Paraschivoiu, I., "Validation and Verification of Multi-Steps Icing Calculation Using CANICE2D-NS Code," 31st AIAA Applied Aerodynamics Conference, AIAA 2013-2671, 2013, doi:10.2514/6.2013-2671.

[17] Smith, A. G. and Spalding, D. B., "Heat transfer in a laminar boundary layer with constant fluid properties and constant wall temperature," *Journal of the Royal Aeronautical Society*, Vol. 62, No. 565, 1958, pp. 60–64,

doi:10.1017/S0368393100067948.

[18] Kerho, M. F. and Bragg, M. B., "Airfoil Boundary-Layer Development and Transition with Large Leading-Edge Roughness," *AIAA Journal*, Vol. 35, No. 1, 1997, pp. 75–84,

doi:10.2514/2.65.

[19] Makkonen, L., "Heat transfer and icing of a rough cylinder," *Cold regions science and technology*, Vol. 10, No. 2, 1985, pp. 105–116,

doi:10.1016/0165-232X(85)90022-9.

[20] Kays, W. M. and Crawford, M. E., *Convective heat and mass transfer*, McGraw-Hill, 1993.

[21] Rafael, C. F., Pio, D. M., and da Silva, G. A. L., "CFD and Boundary Layer Models with Laminar-Turbulent Transition around Airfoils and a Rough Cylinder: Results Validation," SAE Technical Paper, 2015,

doi:10.4271/2015-01-2163.

[22] Radenac, E., Kontogiannis, A., Bayeux, C., and Villedieu, P., "An extended rough-wall model for an integral boundary layer model intended for ice accretion calculations," 10th AIAA Atmospheric and Space Environments Conference - AVIATION 2018 ATLANTA, USA, AIAA 2018-2858, 2018,

doi:10.2514/6.2018-2858.

[23] McClain, S., Vargas, M., Tsao, J.-C., Broeren, A., and Lee, S., "Ice accretion roughness measurements and modeling," 7th European Conference for Aeronautics and Space Sciences (EUCASS), EUCASS2017-555, 2017,

doi:10.13009/EUCASS2017-555.

[24] Han, Y. and Palacios, J., "Surface roughness and heat transfer improved predictions for aircraft ice-accretion modeling," *AIAA Journal*, Vol. 55, No. 4, 2017, pp. 1318–1331,

doi:10.2514/1.J055217.

[25] Wright, W., "A Revised Validation Process for Ice Accretion Codes," 9th AIAA Atmospheric and Space Environments Conference - AVIATION 2017 DENVER, USA, AIAA 2017-3415, 2017,

doi:10.2514/6.2017-3415.

[26] Geuzaine, C. and Remacle, J.-F., "GMSH: A 3-D Finite Element Mesh Generator with Built-in Pre-and Post-Processing Facilities," *International Journal for Numerical Methods in Engineering*, Vol. 79, No. 11, 2009, pp. 1309–1331,

doi:10.1002/nme.2579.

[27] Zienkiewicz, O. C. and Phillips, D. V., "An automatic mesh generation scheme for plane and curved surfaces by isoparametric coordinates," *International Journal for Numerical Methods in Engineering*, Vol. 3, No. 4, 1971, pp. 519–528,

doi:10.1002/nme.1620030407.

[28] Steger, J. and Chaussee, D., "Generation of body-fitted coordinates using hyperbolic partial differential equations," *SIAM J Sci Stat Comput*, Vol. 1, No. 4, 1980, pp. 431–437,

doi:10.1137/0901031.