



HAL
open science

A Galerkin Method for the Simulation of Laminar Boundary Layers on Heated Walls

Emmanuel Radenac, Rémi Harry, Philippe Villedieu

► **To cite this version:**

Emmanuel Radenac, Rémi Harry, Philippe Villedieu. A Galerkin Method for the Simulation of Laminar Boundary Layers on Heated Walls. *Energies*, 2022, 15 (9), pp.3267. <10.3390/en15093267>. <hal-03694357>

HAL Id: hal-03694357

<https://hal.science/hal-03694357v1>

Submitted on 13 Jun 2022

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

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



Distributed under a Creative Commons CC BY 4.0 - Attribution - International License

Article

A Galerkin Method for the Simulation of Laminar Boundary Layers on Heated Walls

Emmanuel Radenac *, Rémi Harry and Philippe Villedieu

ONERA/DMPE, Université de Toulouse, F-31055 Toulouse, France; remi.harry@onera.fr (R.H.); philippe.villedieu@onera.fr (P.V.)

* Correspondence: emmanuel.radenac@onera.fr

Abstract: This paper presents a new solution method for the calculation of laminar thermal boundary layers. The method consists of a coupling between a modal method (Galerkin method) in the direction normal to the wall and a finite volume method in the direction(s) tangential to the wall. It is similar to an integral method in the sense that only a surface mesh is required and that the unknowns are integral quantities (corresponding to the moments up to a fixed order of the temperature profile in the direction normal to the wall). A specificity of the Galerkin method used is that the domain over which the integrals are computed has a variable size that is also an unknown of the problem. Using a series of numerical tests (representative of situations that can be encountered in aeronautics in the case of a wing equipped with a thermal ice protection system), we show that the new method allows us to predict the quantities of interest with a maximum error of a few percent, while a usual integral method (with only one unknown per mesh cell) is unable to treat the case of boundary layers on heated walls with a strong longitudinal temperature gradient, as shown in the literature.

Keywords: thermal boundary layer; integral boundary layer method; Galerkin method; finite volume method; wedge flows



Citation: Radenac, E.; Harry, R.; Villedieu, P. A Galerkin Method for the Simulation of Laminar Boundary Layers on Heated Walls. *Energies* **2022**, *15*, 3267. <https://doi.org/10.3390/en15093267>

Academic Editors: François Morency and Héloïse Beaugendre

Received: 2 March 2022

Accepted: 27 April 2022

Published: 29 April 2022

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2022 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Although the resolution of thermal boundary layers most often relies on solving Navier–Stokes equations, there are still fields of application for which the pursuit of very fast methods of solving this problem is privileged. For instance, aircraft in-flight icing prediction is one field for which it has remained common practice to calculate convective heat transfer 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 airfoil or obstacle is updated. The flow needs to be systematically calculated around this updated shape. Therefore, many aerodynamic calculations may be necessary. The same issue is faced when modeling the activation of thermal ice protection systems (electro-thermal or bleed-air systems): the resolutions of the aerodynamics, of the heat transfer in the substrate of the protected feature and of the water runback and freezing over the protected surface have to be coupled. This also leads to a huge number of aerodynamic simulations. As a result, a robust and fast calculation method for convective heat transfer is interesting for in-flight icing applications. This is the case in 2D but even more so in 3D due to the large number of degrees of freedom.

Icing problems are characterized by a separation of time scales such that the transient state of the aerodynamic flow can be considered as infinitely fast compared to the ice-growth duration and to the time required to reach thermal equilibrium inside the protected airfoil. This justifies the aforementioned use of a sequential coupling of several solvers for the simulations of ice accretion. Steady-state solutions of the aerodynamic field, and especially of the boundary layer, are also wanted for the same reason. To do so, integral-boundary-layer methods are often used for ice-accretion simulations. Since the core of

this paper is dedicated to the modeling of boundary layers over heated walls, it is worth mentioning that boundary-layer solvers are also widely used in the icing community for modeling thermal ice protection systems. For instance, simplified (algebraic) integral methods are used by Morency et al. [1], Al-Khalil et al. [2], Silva et al. [3], Wright [4] and Bu et al. [5]. There are issues of inaccuracy in using such simple approaches on heated walls, as mentioned by Morency et al. [6]. To overcome these issues, it is possible to improve the heated boundary-layer resolution by solving the Navier–Stokes equations, as proposed, for instance, by Croce et al. [7], Wright [4] or Bu et al. [8]. It is also possible to replace the integral boundary-layer solver with a solver of the Prandtl boundary-layer equations as performed at ONERA [9,10] with the solver CLICET [11], or by Morency [6]. The latter approach is consistent with the search for codes with low-CPU time consumption. However, for such an approach, the surface grid is not independent of the location of the stagnation point or separation line, which is a constraint for the ice-accretion-simulation workflow, especially in 3D. Additionally, no solution can be produced by such a solver beyond the separation of the boundary layer. Bayeux et al. [12] thus developed an integral-boundary-layer solver in order to solve the dynamic boundary layer (not the simplified algebraic equations), with the purpose of an easy enough extension to 3D solutions on any kind of unstructured grids [13]. The solver was used successfully for ice-accretion simulations by Radenac et al. [14]. The laminar thermal boundary layer was solved in a similar integral way in [15] under the assumption that the wall temperature is uniform. However, Harry et al. [16] showed that this basic one-equation thermal boundary-layer integral method fails at capturing the evolution of the wall temperature for severe wall-heating conditions. Difficulties arise especially for large variations in wall temperature, which is the actual situation for surfaces thermally protected against ice. Complex temperature profiles are indeed impossible to predict with the one-equation integral method.

One solution to this issue is to use a system of integral equations [16] (a multi-equation model) instead of the usual single-equation method so that history effects can be taken into account through additional degrees of freedom. As the technique proposed in this work for deriving additional equations is to use a Galerkin-type approximation in the direction normal to the wall, it is worth mentioning Graebel et al.'s article [17] concerning the use of such a method for the solution of the thermal boundary layer. Their paper addresses the resolution of both the dynamic and thermal steady-state, incompressible, laminar boundary layers in 2D. The velocity and temperature profiles in the direction normal to the wall are projected onto orthonormal bases of functions, based on Laguerre polynomials. The system of partial differential equations that governs the boundary-layer behavior then becomes a system of ordinary differential equations with respect to the streamwise coordinate. This system is solved by a so-called predictor–corrector approach. Additionally, specific functions are introduced by Graebel et al. in order to speed up the convergence of the resolution. However, it has to be mentioned that the solved system of equations is actually a reshaped set of equations after Falkner–Skan transformations, which are well-adapted to the 2D incompressible laminar regime. In addition, their approach was only validated on a test case where the wall temperature is constant.

The aim of the present paper is therefore to present and discuss a multi-equation method based on an integral approach for the solution of the steady-state energy equation in the boundary layer. The method has to be compliant with simulations of non-uniformly heated walls in order to be able to model the use of ice protection systems. As in Graebel et al.'s article [17], attention is limited to the 2D laminar flow regime that is important for icing applications, as the flow near the leading edge is generally laminar. However, it is required that the model should be compliant with future extensions to 3D configurations and to the turbulent regime (meaning that, for instance, no Falkner–Skan transformation has to be introduced). To meet these constraints, the classical boundary-layer energy equation is used to derive our system of integral equations by applying the Galerkin method with respect to the y -variable (normal direction to the wall). Since this equation is a scalar equation, with no specific 3D effects, the extension to 3D is expected

to be straightforward. Contrary to Graebel et al.'s article [17], the solution of the sole energy equation in the boundary layer will thus be addressed. The solution of the dynamic boundary layer will be supposed to be known and given by another solver in a decoupled manner, which is suitable for incompressible flows. However, since a variable field of temperature in the boundary layer makes the density field variable too, the assumption of constant density will not be made in the derivation of the equations, and the impact of compressibility effects will be discussed. The method presented here is not limited to aircraft in-flight icing applications. In particular, since integral boundary layer methods have already been developed to study wind turbines [18], and these can also be thermally protected against icing [19], the method presented here can also be applied in this area. Integral boundary layer methods may also be used for aerodynamic design in the aeronautical industry [20] or for the modeling of wall mass transfers [21] (and more generally for the rapid design and optimization of any system involving heat and mass transfer induced by forced convection). Nevertheless, in the present work, the method will be tested in representative cases of icing applications only.

The paper is divided into four parts. The first part recalls the boundary-layer energy equation and the corresponding assumptions on which the proposed solution method is based (Section 2). The solution method is shown in Section 3. This is an original method where the evolution in the normal direction to the wall is discretized with a Galerkin method (yielding to a system of integral equations), whereas the streamwise evolution is discretized with the finite volume method. The basis of functions used with the Galerkin method is less optimized for laminar regime than Graebel et al.'s approach [17]. However, it should be possible to extend the method to the turbulent regime in the future (with several adjustments that are not discussed in this article). The last part of the article discusses the use of the proposed method on test cases representative of in-flight icing conditions, especially with non-uniform wall temperature, which are rarely addressed with integral methods in the literature (Section 4).

2. Laminar Boundary-Layer Energy Equation

The goal of this work is to solve an energy equation for the 2D compressible laminar boundary layer, assuming that the flow is steady outside of the boundary layer. The major point of the compressible assumption is to allow the density to be variable, which is mainly due to the dilatability in the case treated here.

Let us consider the variation in the enthalpy $h(t, x, y)$ as a function of time (t) and position in a 2D boundary layer, where x is the streamwise direction and y is the direction normal to the wall. The equation reads [22]:

$$\rho \partial_t(h) + \rho u_x \partial_x(h) + \rho u_y \partial_y(h) = \partial_t(p) + u_x \partial_x(p) - \partial_y(\phi) + \dot{D} \quad (1)$$

where $\rho(t, x, y)$ is the density, $u_x(t, x, y)$ and $u_y(t, x, y)$ are the flow velocities in the x and y direction, respectively, and $p(t, x, y)$ is the pressure. Moreover, the heat flux reads:

$$\phi = -k \partial_y(T) \quad (2)$$

where $T(t, x, y)$ is the temperature and k is the thermal conductivity of the fluid (generally a function of T). Finally, the viscous dissipation reads:

$$\dot{D} = \tau_{xy} \partial_y(u_x) \quad (3)$$

The viscous shear stress is $\tau_{xy} = \mu \partial_y(u_x)$, where μ is the dynamic viscosity of the fluid (also depending on T in general).

Additionally, it is worth recalling that the other equations governing the 2D compressible laminar boundary layer are as follows [22]:

$$\partial_t(\rho) + \partial_x(\rho u_x) + \partial_y(\rho u_y) = 0 \quad (4)$$

$$\rho \partial_t(u_x) + \rho u_x \partial_x(u_x) + \rho u_y \partial_y(u_x) = -\partial_x(p) + \partial_y(\tau_{xy}) \quad (5)$$

$$\partial_y(p) = 0 \quad (6)$$

Equation (4) is the continuity equation, while Equations (5) and (6) are the momentum equations in the streamwise and normal directions, respectively.

For the numerical integration presented in Section 3, it is of practical interest to derive an equivalent equation for the thermal boundary layer by replacing T with a new unknown $\theta(t, x, y) = T_e(x) - T(t, x, y)$, which by definition vanishes at the edge of the boundary layer (where the temperature is T_e). It is worth noticing that the temperature T_e depends only on x , as do the other variables at the edge of the boundary layer, pressure p_e and velocity $u_{x,e}$, because a steady state is assumed and x is the streamwise direction.

First, the following equation for T is derived from Equation (1):

$$\rho c_p \partial_t(T) + \rho c_p u_x \partial_x(T) + \rho c_p u_y \partial_y(T) = u_x d_x(p_e) + \partial_y(k \partial_y(T)) + \dot{D}, \quad (7)$$

which stems from the fact that the specific heat capacity at a constant pressure is $c_p = (\partial_T(h))_p$. This equation also accounts for the fact that Equation (6) implies that $p(t, x, y) = p_e(x)$.

Equation (4) allows to further reshape Equation (7):

$$c_p \partial_t(\rho T) + c_p \partial_x(\rho u_x T) + c_p \partial_y(\rho u_y T) = u_x d_x(p_e) + \partial_y(k \partial_y(T)) + \dot{D} \quad (8)$$

Since the temperature $T_e(x)$ at the edge of the boundary layer depends only on x , the following equation on $T - T_e = -\theta$ is obtained by simply subtracting $T_e \times$ Equation (4) from Equation (8).

$$c_p \partial_t(\rho(T - T_e)) + c_p \partial_x(\rho u_x(T - T_e)) + c_p \partial_y(\rho u_y(T - T_e)) + c_p \rho u_x d_x(T_e) = \begin{matrix} u_x d_x(p_e) \\ + \partial_y(k \partial_y(T - T_e)) \\ + \dot{D} \end{matrix} \quad (9)$$

The terms linked to the derivatives of p_e and T_e can be further simplified because the flow outside of the boundary layer is assumed to be inviscid and steady. Under these assumptions, the x-momentum equation of the system of Euler equations simply reads:

$$\rho_e u_{x,e} d_x(u_{x,e}) = -d_x(p_e) \quad (10)$$

Considering that the total enthalpy $h_{i,e}$ outside of the boundary layer is constant, it can also be stated that:

$$d_x(h_{i,e}) = 0 = c_p d_x(T_e) + \frac{1}{2} d_x(u_{x,e}^2) \quad (11)$$

Finally, Equations (10) and (11) imply that:

$$d_x(p_e) = -\frac{\rho_e}{2} d_x(u_{x,e}^2) = \rho_e c_p d_x(T_e) \quad (12)$$

Consequently, Equation (9) writes:

$$c_p \partial_t(\rho \theta) + c_p \partial_x(\rho u_x \theta) + c_p \partial_y(\rho u_y \theta) = (\rho c_p - \rho_e c_p) u_x d_x(T_e) + \partial_y(k \partial_y(\theta)) - \dot{D} \quad (13)$$

Eventually, if the specific heat capacity is assumed constant, which is a fairly good assumption for air in the relatively narrow range of temperatures investigated in the field of icing (around 223 K to 373 K), the equation governing the relative temperature θ in the boundary layer reads:

$$\partial_t(\rho c_p \theta) + \partial_x(\rho u_x c_p \theta) + \partial_y(\rho u_y c_p \theta) = (\rho - \rho_e) c_p u_x d_x(T_e) + \partial_y(k \partial_y(\theta)) - \dot{D} \quad (14)$$

Although Equation (14) is unsteady, it will be used to build an iterative numerical method that converges towards the steady-state solution. This method is easy to generalize in 3D and on any surfaces. More specifically, the equation that will be solved for this purpose is the following one:

$$\partial_t(\rho_e c_p \theta) + \partial_x(\rho u_x c_p \theta) + \partial_y(\rho u_y c_p \theta) = (\rho - \rho_e) c_p u_x d_x(T_e) + \partial_y(k \partial_y(\theta)) - \dot{D}. \quad (15)$$

Using a density independent of y in the time derivative does not affect the steady-state solution and will simplify the resolution.

There is no major difficulty in deriving the equation for the 3D case, where a crosswise z direction must be defined (the only additional assumption is that of irrotationality of the edge flow). Equation (15) is simply replaced by:

$$\partial_t(\rho_e c_p \theta) + \partial_x(\rho u_x c_p \theta) + \partial_y(\rho u_y c_p \theta) + \partial_z(\rho u_z c_p \theta) = (\rho - \rho_e) c_p (u_x \partial_x(T_e) + u_z \partial_z(T_e)) + \partial_y(k \partial_y(\theta)) - \dot{D} \quad (16)$$

3. Solution Method

The solution method proposed here is a hybrid method of Galerkin type with respect to the y variable and of finite volume type with respect to the other space variables x and z . It has the advantage of requiring only a 2D surface mesh (in 3D) or a 1D linear mesh (in 2D), which not only reduces the computational cost but also greatly simplifies the implementation of the method. Indeed, in practice, the same situation as in the case of an integral method is faced, with the difference that a system of integral equations must be solved (with a dimension depending on the order of the Galerkin method) instead of a single equation for the integral thickness of the boundary layer. In the following, in order to simplify the notations, the method will be presented in the 2D case, but its generalization in 3D does not present any theoretical difficulty.

3.1. Galerkin Method with Respect to the y Variable

The Galerkin method consists of searching for an approximation of the solution $\theta(t, x, y)$ of the following form:

$$\tilde{\theta}(t, x, y) \stackrel{\text{def}}{=} \sum_{j=1}^N \theta_j(t, x) \varphi_j\left(\frac{y}{\delta_T(t, x)}\right) \quad (17)$$

where δ_T denotes the thickness of the thermal boundary layer, θ_j are the modal coordinates (all having the dimension of a temperature) and the φ_j are the basis functions defined as:

$$\begin{cases} \varphi_j(\hat{y}) = \pi_j(\hat{y})(1 - \hat{y})^q & \text{if } \hat{y} \in [0, 1], \\ \varphi_j(\hat{y}) = 0 & \text{if } \hat{y} \in [1, +\infty[. \end{cases} \quad (18)$$

where $\hat{y} = \frac{y}{\delta_T}$, q is a strictly positive constant and π_1, \dots, π_N are N independent polynomials, whose expressions will be specified later (the degree of π_j is less than or equal to j). It is worth mentioning that the approximate solution $\tilde{\theta}(t, x, y)$ satisfies by construction the boundary condition $\theta(t, x, \delta_T(t, x)) = 0$. As N increases, $\tilde{\theta}$ is a polynomial of increasingly higher degree in y and thus tends to oscillate in the vicinity of $y = \delta_T$. However, since the integer part of $q - 1$ corresponds to the number of zero derivatives of $\tilde{\theta}$ at $y = \delta_T$, q allows us to control the behaviour of the solution in the vicinity of the boundary layer edge. At fixed N , if q is large enough, the oscillations can be suppressed, but if q is too high, the solution tends to flatten in the neighborhood of $y = \delta_T$, which leads to a decrease in the accuracy of the method. After various numerical tests, the choice $q = \max(2, N)$ appeared to be a good compromise and this value was thus adopted for all the numerical tests, which will be presented later in the paper.

From Equations (17) and (18), it appears that, at fixed t and x , the approximate solution $\tilde{\theta}$ is determined by $N + 1$ independent parameters, the modal coordinates $\theta_1, \dots, \theta_N$ and

the thermal boundary layer (approximate) thickness δ_T . One of these degrees of freedom can be easily removed. Indeed, using the Dirichlet boundary condition, $\theta = \theta_w \stackrel{\text{def}}{=} T_e - T_w$ in $y = 0$, the following relationship is obtained:

$$\sum_{j=1}^N \theta_j \varphi_j(0) = \theta_w \tag{19}$$

Hence, if the π_j polynomials are chosen such that $\pi_1(0) = 1$ and $\pi_j(0) = 0$ for $j > 1$, this equation greatly simplifies:

$$\theta_1 = \theta_w \tag{20}$$

The N unknowns of the problem are thus finally $\theta_2, \dots, \theta_N, \delta_T$, and N independent equations must now be derived from the master Equation (15). To do this, Equation (15) just needs to be projected onto the space \mathcal{P}_{N-1} of polynomials in y of degree less than or equal to $N - 1$. The solution of Equation (15) must therefore verify:

$$\forall P \in \mathcal{P}_{N-1}, \int_0^{+\infty} [\partial_t(\rho_e c_p \theta) + \partial_x(\rho c_p u_x \theta) + \partial_y(\rho c_p u_y \theta) - \partial_y(k \partial_y \theta)] - (\rho - \rho_e) c_p u_x \partial_x T_e + \dot{D} P(y) dy = 0 \tag{21}$$

\mathcal{P}_{N-1} is a vector space of dimension N ; it suffices to take for P the N monomials ψ_i defined by $\psi_i(y) \stackrel{\text{def}}{=} C_i y^{i-1}$ for $i = 1 \dots N$, where the C_i are normalization constants whose values will be specified later on. By replacing in Equation (21), the exact solution θ by the approximate solution $\tilde{\theta}$ and the polynomial P by each of the monomials ψ_i , the following system of equations is obtained:

$$\forall i = 1 \dots N, \quad \partial_t M_i + \partial_x F_{x,i} = S_i \tag{22}$$

where by definition (the variables x and t being omitted for clarity):

$$M_i \stackrel{\text{def}}{=} \rho_e c_p \int_0^{+\infty} \tilde{\theta}(y) \psi_i(y) dy \tag{23}$$

$$F_{x,i} \stackrel{\text{def}}{=} \int_0^{+\infty} \rho(y) c_p u_x(y) \tilde{\theta}(y) \psi_i(y) dy \tag{24}$$

$$S_i \stackrel{\text{def}}{=} \int_0^{+\infty} \psi_i(y) \partial_y(k \partial_y \tilde{\theta})(y) dy - \int_0^{+\infty} \partial_y(\rho c_p u_y \tilde{\theta})(y) \psi_i(y) dy + c_p \partial_x T_e \int_0^{+\infty} (\rho(y) - \rho_e) u_x(y) \psi_i(y) dy - \int_0^{+\infty} \dot{D}(y) \psi_i(y) dy \tag{25}$$

It can be seen that the application of the Galerkin’s method with respect to the variable y leads to a system of conservation laws whose unknowns are the first N moments of the temperature profile (up to a multiplicative constant) and whose terms $F_{x,i}$ and S_i correspond respectively to the convective fluxes in the x direction and to the source terms. The physical meaning of the different terms composing S_i will be discussed later on.

For the system (22) to be mathematically well-posed, the relation between the moments M_i ($i = 1 \dots N$) and the temperature profile $\tilde{\theta}$ must be bijective so that the terms $F_{x,i}$ and S_i can be considered as implicit functions of the moments. The necessity of this condition will become clear in the next subsection dedicated to the finite volume discretization of the system (22). One must therefore address the question of the computation of θ_j for $j = 1 \dots N$ and δ_T from the first N moments of $\tilde{\theta}$, i.e., the question of the inversion of the system corresponding to Equation (23) for i varying from 1 to N . For this purpose, using the expression of $\tilde{\theta}$, Equation (23) can be rewritten in the following matrix form:

$$M_i = \sum_{j=1}^N A_{ij}(\delta_T) \theta_j \tag{26}$$

where the coefficients of the matrix $A_{ij}(\delta_T)$ are given by:

$$A_{ij}(\delta_T) \stackrel{\text{def}}{=} \int_0^{+\infty} \rho_e c_p \varphi_j \left(\frac{y}{\delta_T} \right) \psi_i(y) dy = C_i \delta_T^i \rho_e c_p \int_0^1 \pi_j(\hat{y}) (1 - \hat{y})^q \hat{y}^{i-1} d\hat{y} \quad (27)$$

From the point of view of the finite volume discretization of system (22), which will be exposed in the following section, the computation of δ_T and of the θ_j from the M_i needs to be simple and always possible. For this, it is possible to play on the choice of the polynomials π_j . Let us set by definition:

$$\pi_j(\hat{y}) = B_{j0}\hat{y}^0 + B_{j1}\hat{y}^1 + B_{j2}\hat{y}^2 + \dots + B_{jj}\hat{y}^j \quad (28)$$

For the polynomial of the lowest degree, π_1 , it is natural to take a constant polynomial and thus to set $\pi_1 \equiv 1$ (since all polynomials can be normalized arbitrarily). For the other polynomials ($j > 1$), it was shown that it is necessary to have $\pi_j(0) = 0$ to obtain the simple relation $\theta_1 = \theta_w$. So, for all $j = 2 \dots N$, the polynomials π_j read:

$$\pi_j(\hat{y}) = B_{j1}\hat{y}^1 + B_{j2}\hat{y}^2 + \dots + B_{jj}\hat{y}^j \quad (29)$$

At fixed j , j conditions have to be imposed in order to fix the values of the coefficients B_{jk} . We have chosen to impose that the matrix A_{ij} is lower triangular (which ensures the possibility to inverse the system (26)) and that its diagonal terms verify the normalization condition $A_{ii} = \delta_T^i \rho_e c_p$. At fixed j , this leads to the following linear system of unknowns $(B_{jk})_{k=1\dots j}$:

$$\begin{cases} \sum_{k=1}^j \left(\int_0^1 (1 - \hat{y})^q \hat{y}^{i-1+k} d\hat{y} \right) B_{jk} = 0, & \forall i = 1, \dots, j-1, \\ \sum_{k=1}^j \left(\int_0^1 (1 - \hat{y})^q \hat{y}^{j-1+k} d\hat{y} \right) B_{jk} = \frac{1}{C_j} \end{cases} \quad (30)$$

The coefficients of this system $\Pi_{ik} = \int_0^1 (1 - \hat{y})^q \hat{y}^{i-1+k} d\hat{y}$ can be easily calculated by recurrence, leading to:

$$\Pi_{ik} = \frac{q!(k+i-1)!}{(q+k+i)!} \quad (31)$$

At a fixed k , they are all the smaller as q and i are large. To make sure that all the B_{jk} are of the same order of magnitude whatever the value of j between 2 and N , we have chosen to set:

$$C_j = \left(\int_0^1 (1 - \hat{y})^q \hat{y}^{j-1} d\hat{y} \right)^{-1} = \frac{(q+j)!}{q!(j-1)!} \quad (32)$$

It is worth noticing that at the continuous level (or if a computer of infinite precision is used for the computations), the choice of C_i has no incidence on the values of the approximated solution $\tilde{\theta}$ since only the products $\theta_j \pi_j$ play a role. The larger π_j is, the smaller θ_j is. The normalization is only important in finite precision because it avoids a too-large disparity between the θ_j and thus allows a better conditioning of the problem.

For all $j = 2 \dots N$, the system (30) can be solved analytically, but this is not useful as its numerical solution is very fast and does not present any difficulties (except for large j since the system then becomes ill-conditioned). In practice, the system (30) is solved numerically once and for all when the solver is launched and the values of N and q are fixed.

Let us now return to the problem of computing δ_T and θ_i , knowing the M_i . It has already been determined that $\theta_1 = \theta_w$. In addition, since by construction $A_{1j} = 0$ for all $j > 1$, the first equation of system (26) implies:

$$M_1 = A_{11}\theta_1 \quad (33)$$

with:

$$A_{11} = C_1 \delta_T \rho_e c_p \int_0^1 (1 - \hat{y})^q d\hat{y} = \delta_T \rho_e c_p \quad (34)$$

Hence:

$$\delta_T = \frac{M_1}{\rho_e c_p \theta_w} \quad (35)$$

δ_T being known thanks to Equation (35) and the matrix A_{ij} being lower triangular, the computation of the θ_i for $i = 2 \dots N$ is then straightforward.

Finally, let us give some details on the physical meaning and the computation of the source terms S_i ($i = 1 \dots N$), which are actually composed of four terms according to Equation (25). The first term of S_i :

$$\Phi_i(\tilde{\theta}) \stackrel{\text{def}}{=} \int_0^{+\infty} \psi_i(y) \partial_y (k \partial_y \tilde{\theta})(y) dy$$

accounts for the effect of the conductive heat transport in the y direction. Integrating by part and using the expression of $\tilde{\theta}$, it can be rewritten as:

$$\Phi_i(\tilde{\theta}) = - \sum_{j=1}^N \theta_j \int_0^{+\infty} \frac{k}{\delta_T} \psi_i'(y) \varphi_j' \left(\frac{y}{\delta_T} \right) dy - \sum_{j=1}^N \theta_j \frac{k(0)}{\delta_T} \varphi_j'(0) \psi_i(0) \quad (36)$$

which leads to:

$$\begin{cases} \Phi_1(\tilde{\theta}) = -k(0) C_1 \delta_T^{-1} \sum_{j=1}^N \theta_j \varphi_j'(0) & i = 1 \\ \Phi_i(\tilde{\theta}) = -C_i (i-1) \delta_T^{i-2} \sum_{j=1}^N \theta_j \int_0^1 k(\hat{y}) \hat{y}^{i-2} \varphi_j'(\hat{y}) d\hat{y} & i = 2 \dots N \end{cases} \quad (37)$$

which can also be written in the following matrix form:

$$\Phi_i(\tilde{\theta}) = \sum_{j=1}^N K_{ij}(\tilde{\theta}) \theta_j \quad (38)$$

The second term of S_i :

$$G_i(\tilde{\theta}) = - \int_0^{+\infty} \partial_y (\rho c_p u_y \tilde{\theta})(y) \psi_i(y) dy$$

corresponds to the convective heat flux in the y direction. Performing an integration by part, the term G_i reads:

$$G_i(\tilde{\theta}) = (i-1) C_i \int_0^{+\infty} \rho(y) c_p u_y(y) \tilde{\theta}(y) y^{i-2} dy \quad (39)$$

The third term of S_i :

$$R_i(\tilde{\theta}) = c_p \partial_x T_e \int_0^{+\infty} (\rho(y) - \rho_e) u_x(y) \psi_i(y) dy \quad (40)$$

takes into account the influence of the density variations across the boundary layer. In this equation and the previous one, the density profile is related to the temperature profile using the relation:

$$\rho(y) = \frac{p_e}{rT(y)} = \frac{p_e}{r(T_e - \tilde{\theta}(y))} \quad (41)$$

Lastly, the fourth term of S_i :

$$\dot{Q}_i = - \int_0^{+\infty} \dot{D}(y)\psi_i(y)dy$$

accounts for the heat production due to viscous forces.

It should be noted that for the extension to 3D, the only changes are the following. Equation (16) is used as the starting point for the analysis, instead of Equation (15). Consequently, the system to be solved becomes:

$$\forall i = 1 \dots N, \quad \partial_t M_i + \partial_x F_{x,i} + \partial_z F_{z,i} = S_i$$

where there is a cross-wise component $F_{z,i}$ to the convective flux term:

$$F_{z,i} \stackrel{\text{def}}{=} \int_0^{+\infty} \rho(y)c_p u_z(y)\tilde{\theta}(y)\psi_i(y)dy.$$

Only the third term of S_i , R_i , has to be modified to account for the edge-temperature gradient along the cross-wise direction:

$$R_i(\tilde{\theta}) = c_p \partial_x T_e \int_0^{+\infty} (\rho(y) - \rho_e)u_x(y)\psi_i(y)dy + c_p \partial_z T_e \int_0^{+\infty} (\rho(y) - \rho_e)u_z(y)\psi_i(y)dy.$$

3.2. Finite Volume Method with Respect to the x Variable

To simplify the notations it is useful to set:

$$\mathbf{M} = \begin{pmatrix} M_1 \\ \vdots \\ M_N \end{pmatrix}, \quad \mathbf{F} = \begin{pmatrix} F_{x,1} \\ \vdots \\ F_{x,N} \end{pmatrix}, \quad \mathbf{S} = \begin{pmatrix} S_1 \\ \vdots \\ S_N \end{pmatrix}, \quad \Theta = \begin{pmatrix} \theta_1 \\ \vdots \\ \theta_N \end{pmatrix}$$

With these notations, the system (22) can be rewritten in the following more compact form:

$$\partial_t(\mathbf{M}) + \partial_x(\mathbf{F}(\Theta, \delta_T)) = \mathbf{S}(\Theta, \delta_T) \tag{42}$$

The finite volume method and the explicit Euler scheme applied to Equation (42) lead to the following numerical scheme:

$$\mathbf{M}_i^{n+1} = \mathbf{M}_i^n - \frac{\Delta t}{\Delta x_i} (\mathbf{F}_{i+1/2}^n - \mathbf{F}_{i-1/2}^n) + \Delta t S_i^n \tag{43}$$

where by definition Δt is the time step, \mathbf{M}_i^n denotes the value of the discrete solution in the mesh cell $C_i =]x_{i-1/2}, x_{i+1/2}[$ at time $t_n = n\Delta t$, $\Delta x_i = x_{i+1/2} - x_{i-1/2}$ is the size of the cell C_i , $\mathbf{F}_{i+1/2}^n$ is the numerical flux between the cells C_i and C_{i+1} on the time interval $[t_n, t_{n+1}]$, and S_i^n is the discrete value of the source term in the cell C_i on the time interval $[t_n, t_{n+1}]$.

The way to calculate the numerical fluxes and the source terms remains to be specified. As our first objective was simply to verify the potential of our method, we only implemented very simple numerical schemes. On the one hand, an explicit time discretization scheme is used, and on the other hand an upwind first order scheme is applied for the computation of the numerical fluxes (by upwinding according to the sign of the velocity u_{xe}). We have thus taken:

$$\begin{cases} \mathbf{F}_{i+1/2}^n = \mathbf{F}(\Theta_i^n, \delta_{T_i}^n, u_{xe}(x_{i+1/2})) & \text{if } u_{xe}(x_{i+1/2}) > 0 \\ \mathbf{F}_{i+1/2}^n = \mathbf{F}(\Theta_{i+1}^n, \delta_{T_{i+1}}^n, u_{xe}(x_{i+1/2})) & \text{if } u_{xe}(x_{i+1/2}) \leq 0, \end{cases} \tag{44}$$

where the edge velocity used for upwinding is $u_{xe}(x_{i+1/2}) = (u_{xe}(x_i) + u_{xe}(x_{i+1}))/2$, and

$$S_i^n = \mathbf{S}(\Theta_i^n, \delta_{T_i}^n) \tag{45}$$

As they only involve smooth functions, all the integrals contained in the expression of the numerical fluxes and source terms can be approximated by using a Gauss–Legendre quadrature formula with q nodes, which is exact when the function to be integrated is a polynomial of degree $2q - 1$. In practice, we used a formula with $2N$ nodes where N is the number of modes of the Galerkin method. We verified through different tests that the integration error was negligible compared to the other approximation errors.

Finally, the proposed method of resolution comes down to the following algorithm:

- Solution initialization: in each cell, set: $\delta_{Ti}^0 = \delta_D(x_i)$, where δ_D is the dynamic boundary layer thickness and $\theta_{1i}^0 = \theta_w$, $\theta_{2i}^0 = 0, \dots, \theta_{Ni}^0 = 0$.
- Time iteration: in each cell, knowing \mathbf{M}_i^n from the previous time step compute first δ_{Ti}^n and Θ_i^n by using Equations (35) and (26). Then, compute the numerical fluxes and the source terms using Gauss–Legendre quadrature formulas, and finally update the solution using the scheme (43).
- End of the computation: Iterate until convergence to the steady state, i.e., until a given norm of the residual drops below a predefined threshold.

As the objective is only to compute the stationary solution, it is preferable to choose the largest possible time step to accelerate the convergence. However, with the algorithm being explicit, the maximum value of the (local) time step Δt_i is limited by a numerical stability criterion of the form:

$$\text{Max} \left(\frac{|u_{xe,i}| \Delta t_i}{\Delta x_i}, \frac{2\kappa_e N^2 (q+1)^2 \Delta t_i}{(\delta_{Ti}^n)^2} \right) \leq \text{CFL}_{max} \quad (46)$$

where κ_e is the edge value of the thermal diffusivity $\kappa \stackrel{\text{def}}{=} \frac{k}{\rho c_p}$ and CFL_{max} is a numerical parameter that can be taken to be equal to 1 according to all our numerical experiments (in practice, $\text{CFL}_{max} = 0.9$ was used for the simulations of this article). The first term $\frac{|u_{xe,i}| \Delta t_i}{\Delta x_i}$ is related to the convective flux in the x direction, and the second term $\frac{2\kappa_e N^2 (q+1)^2 \Delta t_i}{(\delta_{Ti}^n)^2}$ is related to the heat diffusion source term. The presence of N^2 is due to the fact that the higher the number of modes, the smaller the characteristic diffusion length scale in the y direction, which behaves like $\frac{\delta_T}{N(q+1)}$. It must be noted that $\delta_T / (q+1)$ corresponds to the effective boundary layer thickness rather than δ_T itself. It will indeed be demonstrated later on that, contrary to δ_T , this parameter converges to a constant value when the number of modes N increases. Additionally, the factor 2 is caused by the very usual constraint of a Fourier number criterion being lower than 1/2 in explicit resolution of the heat equation. In future works, the constraint (46) can be easily overcome by using an implicit Euler scheme instead of the current explicit Euler scheme.

Regarding the extension to 3D, as the system to be solved is only slightly modified, the same finite-volume method can be used. A standard upwind numerical scheme for unstructured surface meshes will then naturally handle the two components of the convective flux identified earlier.

4. Results and Discussion

The modal method described in Section 3 has been implemented in a code called BLIM2D. Several test cases computed with BLIM2D will be discussed in this section in order to show the interest of the method.

4.1. Test Case Definition

In this paper, wedge air-flow test cases will be considered. Flat profiles are thus investigated, for which the edge velocity is simply $u_{xe} = Kx^m$ (Figure 1). K and m are two constants that have to be set. K allows us to change the Mach number of the test

case, while m is linked to the pressure gradient (and to the angle of the wedge). As a reminder, the well-known Falkner–Skan solutions are the solutions of the wedge-flow dynamic boundary layer. The wedge-flow cases are interesting because they allow us to validate the proposed solution method for a large set of academic test cases that are representative of real conditions. For instance, $m = 1$ corresponds to the air flow in the vicinity of a stagnation point (wedge with an angle equal to $\pi/2$), which is a key area for icing applications. It is indeed a laminar region where some ice deposition occurs and where thermal protection systems are usually installed and produce a high amount of heat. This explains the large number of cases with this value of m in the remainder of the article. Further downstream, the flow is accelerated, which can be modelled by wedge flows with $m \in [0, 1]$, with $m = 0$ being the flat-plate condition with zero pressure gradient (and constant edge velocity). Further downstream, the flow is finally decelerated, which can be modelled with $m < 0$ wedge-flows.

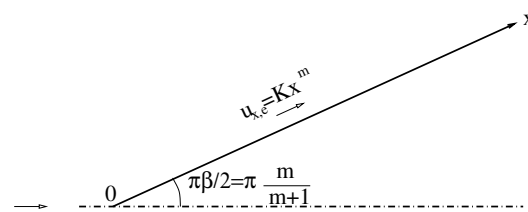


Figure 1. Definition of the wedge-flow geometry.

The wall temperature will be set either constant or linearly varying. The zero-temperature-gradient cases are simple enough cases for a basic single-equation integral-boundary-layer method to produce a good solution [13,16]. However, some aspects regarding the numerical resolution will be specifically investigated in these cases. The expected numerical results will be discussed in Section 4.2.1 for conditions allowing self-similar temperature profiles (test-case 1 of Table 1, as discussed later). The grid convergence of the solution will be shown in Section 4.2.2. The convergence of the solution with the number N of modes will be shown in Section 4.2.3. In particular, the wall heat flux and the temperature profiles produced by BLIM2D will be analyzed.

Regarding the non-zero temperature-gradient cases (Section 4.4), they were chosen after some literature review. In an article by Papadakis et al. [23], the wall temperature was shown to evolve almost linearly in some cases, with temperature gradients close to -400 K/m. The wall temperature is thus warmer at the beginning of the geometry (physical stagnation point) than further downstream. Test case 4 is expected to model these conditions (Table 1). For numerical purposes, test cases where the temperature gradient is zero or positive ($+400$ K/m) are also considered (test cases 3 and 5, respectively).

For these cases, the solution will be compared against a numerical reference solution provided by the code CLICET [11], which is a quite widely used solver for boundary-layer modeling [9,10,24,25]. Indeed, CLICET directly solves the boundary layer equations (Prandtl equations), and especially Equation (1), by using a 2D mesh and a finite volume method. It should be noted that the velocity fields in the boundary layer are not strictly the same for the two codes. BLIM2D is fed with the integral boundary layer solution exposed in the article of Bayeux et al. [12] (see Appendix A for more information), whereas CLICET uses its own velocity field, which is solved jointly with the temperature field. CLICET's velocity field can thus be affected by the low effects of compressibility and moderate effects of dilatability. This potential source of discrepancies between BLIM2D and CLICET will be addressed in Section 4.3. To be more specific, in BLIM2D, the streamwise velocity profile $u_x(y)$ is computed using the integral method described in Bayeux et al. [12], while the normal velocity profile is given by the following equation:

$$u_y(y) = \int_0^y -\partial_x(u_x)(y') dy', \quad (47)$$

under the assumption that the velocity field is incompressible (which is consistent with Bayeux's model assumptions [12]). Bayeux's velocity profile is defined for $0 \leq y \leq \delta$, where δ is the dynamic boundary-layer thickness produced by the dynamic-boundary-layer integral model [12]. For $y > \delta$, the velocity profile is extended with $u_x = u_{xe}$.

Table 1. Physical parameters for the test cases. M is the Mach number at the end of the plate ($x = L$), T_{w0} is the wall temperature at $x = 0$, p_{ie} and T_{ie} are the total pressure and total temperature of the edge flow, respectively.

Test-Case	L (m)	m	M	p_{ie} (Pa)	T_{ie} (K)	T_{w0} (K)	$\vec{\nabla} T_w \cdot \vec{e}_x$ (K/m)
1	0.25	1	0.01	80,005	263.155	264.15	0
2	0.25	−0.01961, 0, 1/3, 1	0.01	80,005	263.155	264.15, 273.15, 323.15, 373.15	0
3	0.25	−0.01961, 0, 1/3, 1	0.25	83,555	266.439	323.15	0
4	0.25	−0.01961, 0, 1/3, 1	0.25	83,555	266.439	373.15	−400
5	0.25	−0.01961, 0, 1/3, 1	0.25	83,555	266.439	273.15	400

It should be noted (see Appendix B) that the model has also been validated against an empirical relation for a non-uniform wall temperature.

4.2. Convergence of the Method

The convergence of the method is assessed by refining the finite-volume grid and by increasing the number of modes of the Galerkin method.

For both investigations, the edge-flow conditions are those of test-case 1 in Table 1: a stagnation point ($m = 1$), with a low Mach number $M = 0.01$ at the end of the plate ($L = 0.25$ m) and a constant wall temperature $T_w = 264.15$ K. As in all the following sections, uniform grids are used for which the number of cells N_x lies between 64 and 512.

The Mach number is very low as well as the difference between the wall temperature and the edge-flow temperature. The very low Mach number implies low dissipation in the boundary layer, whereas the low temperature difference produces low dilatation effects and almost constant density in the boundary layer. The dynamic boundary layer is thus expected to remain unaffected by the wall heating. Both effects are favorable conditions for the self-similar behavior of the thermal boundary layer.

4.2.1. Description of the Investigated Test Case

First of all, the input parameters of simulations are of various kinds: the physical conditions given in Table 1 (boundary conditions at the edge of the boundary layer and at the wall), the velocity field in the boundary layer and the numerical parameters N and N_x . Regarding the velocity field inside the boundary layer (for $N_x = 256$), Figure 2 confirms that the velocity field provided to BLIM2D is not exactly the same as the reference CLICET velocity profile, although the discrepancies are small.

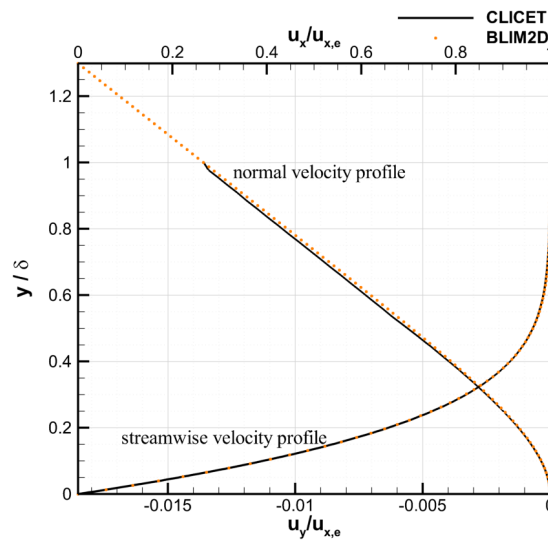


Figure 2. Velocity profiles for CLICET (reference) and BLIM2D, at $x/L = 1$. Test case 1, $N_x = 256$.

For $N_x = 256$ and $N = 5$, Figure 3a shows that for the first equation of system (22), the heat-flux source term Φ_1 is the only source term that does not vanish. For the second equation (Figure 3b), the dissipation \dot{Q}_2 and density R_2 terms still vanish. The same behaviour is observed for the other equations, 3 to N (but it is not shown here for the sake of conciseness). This is consistent with the fact that the test-case running conditions were chosen so that dissipation is negligible and density varies little in order to achieve self-similar temperature profiles. It must be noted that the convective term G_2 is of the same order of magnitude as the heat flux term Φ_2 (although remaining lower). The same behavior is observed for the other equations, 3 to N (however, the source terms become lower for higher number of equations).

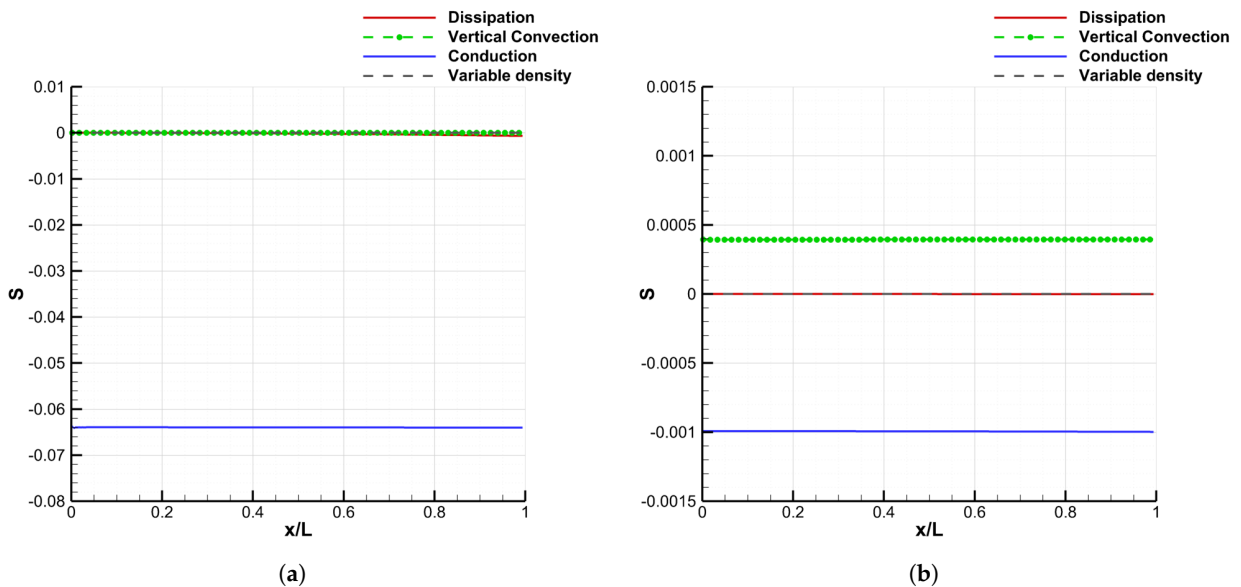


Figure 3. Source terms for $N = 5$. (a) Equation (1), (b) Equation (2). Test-case 1, $N_x = 256$.

Figure 4 confirms that self-similar temperature profiles are produced by BLIM2D, and here for $N = 4$: all the temperature profiles are superimposed at every point of the grid.

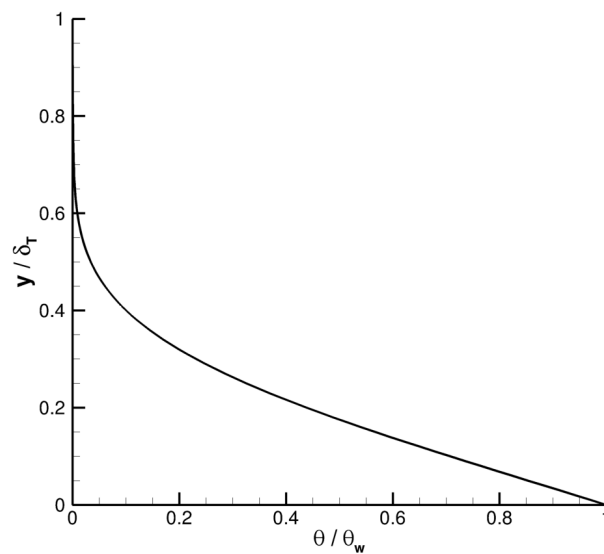


Figure 4. Non-dimensional temperature profiles for $N = 4$ at all the points of the grid. Test case 1, $N_x = 256$.

4.2.2. Grid Convergence

The finite volume method employed for the solution is not the most original part of this article. Therefore, this section only aims at showing that proper convergence is achieved when using refined surface grids. For a resolution with $N = 4$ equations, Figure 5 shows that the heat flux ϕ_w is almost constant. There are some troubles in the vicinity of the inlet boundary condition at $x/L = 0$. The results become smoother there for finer meshes. Otherwise, all mesh sizes $N_x \in \{64, 128, 256, 512\}$ produce similar results. The computation of the relative error in the wall heat flux (compared to the finer mesh) shows that $N_x = 256$ allows errors lower than 10^{-5} for $x/L > 0.1$ (Figure 6).

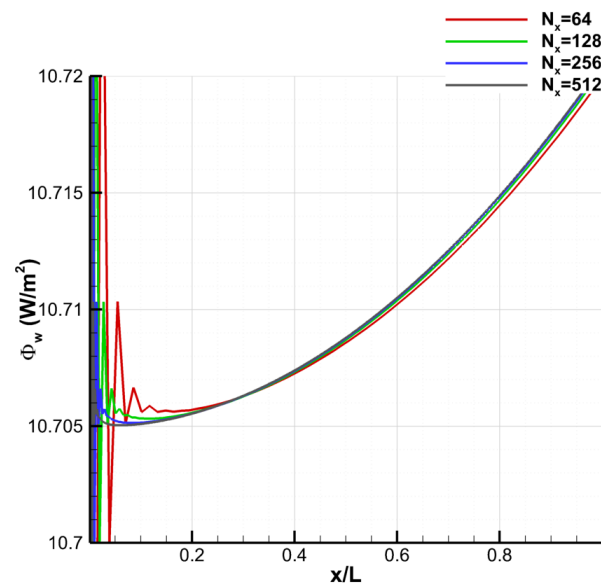


Figure 5. ϕ_w for $N_x = 64$ to $N_x = 512$. Test case 1, $N = 4$.

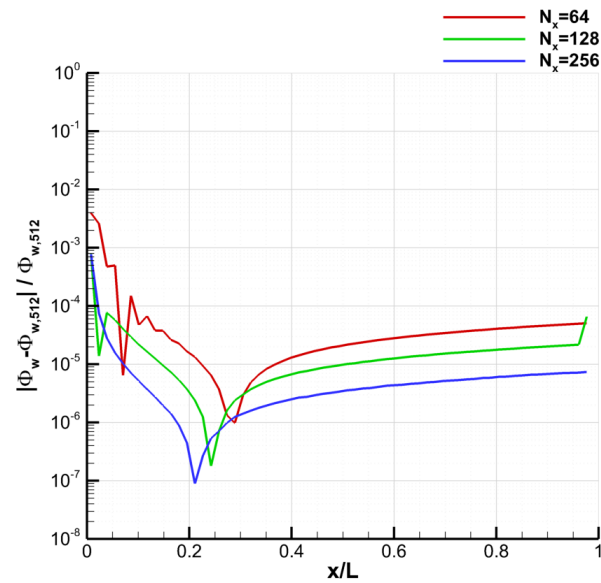


Figure 6. Relative error in ϕ_w for $N_x = 64$ to $N_x = 256$. Test case 1, $N = 4$.

4.2.3. Convergence of the Galerkin Method

For the convergence of the Galerkin method, a uniform grid with $N_x = 256$ points was used. The number of modes is increased from $N = 1$ to $N = 5$ to show the convergence. Since the Galerkin method is the core of this article, it is worth discussing several aspects of the solution that are affected by the number of modes employed.

For the self-similar solution investigated here, where the temperature profiles are the same for all values of x , Equation (17) shows that the weights θ_j are expected to be constant. Figure 7a shows θ_1 , which is strictly equal to θ_w as imposed by Equation (20). Another example is shown with θ_2 in Figure 7b. A constant value is obtained for all values of N (actually, a small variation of less than 1% is obtained), which indicates a good behavior of the solution. However, for a given j , the θ_j values are not expected to converge when N increases. For instance, Figure 7b shows that $|\theta_2|$ grows when N increases (obviously, θ_2 was set to 0 for $N = 1$). The same behavior was observed for θ_3, θ_4 and θ_5 (the small variation can reach a few percent for θ_4 and θ_5), but it is not shown for the sake of conciseness. It must also be mentioned that, for a given value of N , the weights θ_j tend to decrease when j increases. For instance, Figure 7a,b show that θ_1 is equal to θ_w , whereas θ_2/θ_w is of the order of 10^{-2} to 10^{-1} . This reflects the fact that each additional mode contributes to the solution less and less strongly.

As explained in Section 3, the thermal-boundary-layer thickness δ_T is an output of the numerical method. Figure 8 shows that it does not depend on x/L , which is consistent with the fact that the boundary layer thickness is constant for $m = 1$ [22]. The figure also demonstrates that δ_T is not the same for all values of N . This is actually expected because Equation (35) implies that δ_T is proportional to $C_1 = q + 1 = \max(N + 1, 3)$ via M_1 (for self-similar temperature profile and constant δ_T). $\delta_T/(q + 1)$ was thus also plotted in Figure 8 (dashed lines). A non-monotonic convergence is obtained for this parameter, which changes very little. The reference δ_T in Figure 8 was provided by CLICET. The definition of δ_T is actually not the same for CLICET (height at which θ reaches $0.01 \times \theta_w$) and for BLIM2D (height at which the presumed temperature profile of Equation (17) reaches 0). In order to compare between the temperature profiles produced by CLICET and by BLIM2D, it is mandatory to define a common scale for the y coordinate. In the remainder of the article, this scale will be the thickness produced by CLICET, $\delta_{T,ref}$.

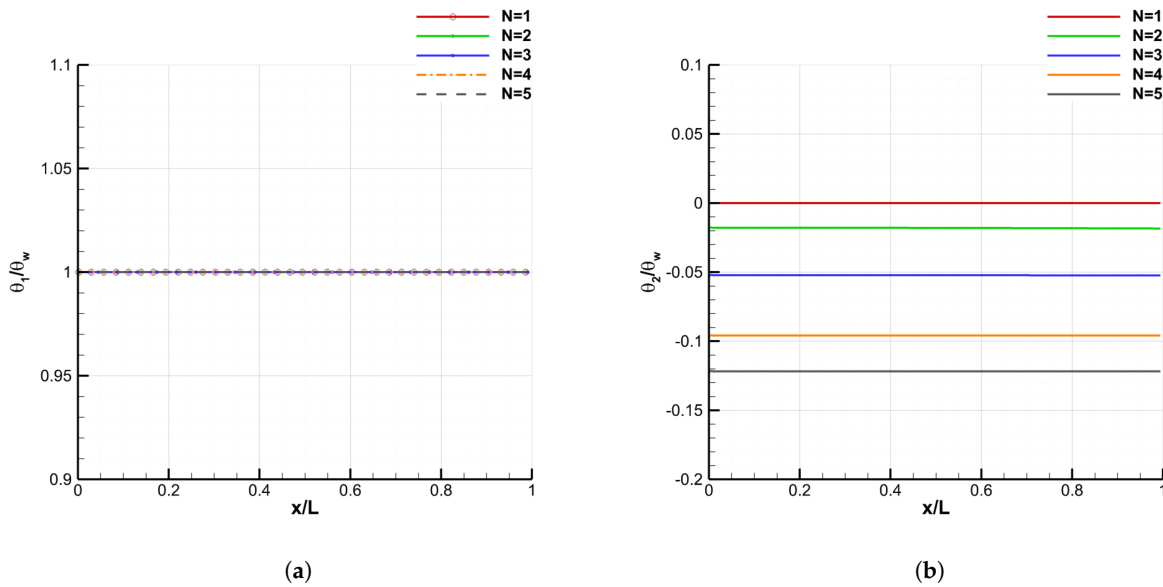


Figure 7. θ_j for $N = 1$ to $N = 5$: (a) θ_1 , (b) θ_2 . Test case 1, $N_x = 256$.

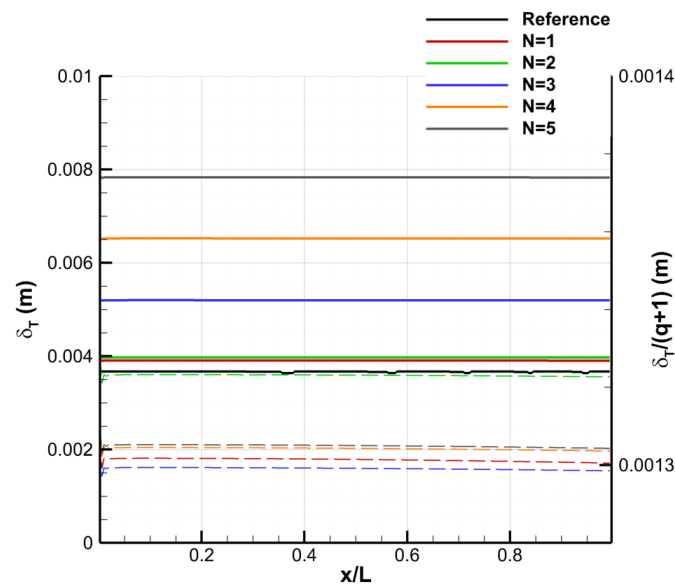


Figure 8. δ_T for $N = 1$ to $N = 5$, and for the reference CLICET solution (solid lines). $\delta_T/(q + 1)$ for $N = 1$ to $N = 5$ (dashed lines). Test case 1, $N_x = 256$.

Figure 9a shows the non-dimensional temperature profiles produced by BLIM2D for $N = 1$ to $N = 5$ modes, at $x/L = 1$ (the temperature profile is self-similar, so any value of x/L gives the same solution, except for small disturbances in the vicinity of $x/L = 0$). The agreement between the BLIM2D solutions and the reference solution is good, especially for $N = 3$ to $N = 5$. To better quantify the errors in the BLIM2D solutions, three error norms defined as follows are plotted in Figure 9b. The L_1 norm is defined as:

$$\epsilon_{L1} = \frac{2}{L} \int_{L/2}^L \frac{1}{\delta_{T,ref}} \int_0^{\delta_{T,ref}} |\theta(x, y) - \theta_{ref}(x, y)| dy dx, \tag{48}$$

the L_2 norm is defined as:

$$\epsilon_{L2} = \left(\frac{2}{L} \int_{L/2}^L \frac{1}{\delta_{T,ref}} \int_0^{\delta_{T,ref}} |\theta(x, y) - \theta_{ref}(x, y)|^2 dy dx \right)^{0.5}, \tag{49}$$

and the the L_∞ norm is defined as:

$$\varepsilon_{L_\infty} = \max_{x \in [L/2, L]} \left(\max_{y \in [0, \delta_{T,ref}]} \left(|\theta(x, y) - \theta_{ref}(x, y)| \right) \right). \tag{50}$$

For all three norms, the error decreases from $N = 2$ to $N = 4$. $N = 5$ does not allow us to make the error drop further. It has to be noted that the L_∞ error is lower than 0.01 K, i.e., lower than 1% of the temperature difference between the wall and the edge of the boundary layer. This is really low for an error in temperature for icing applications.

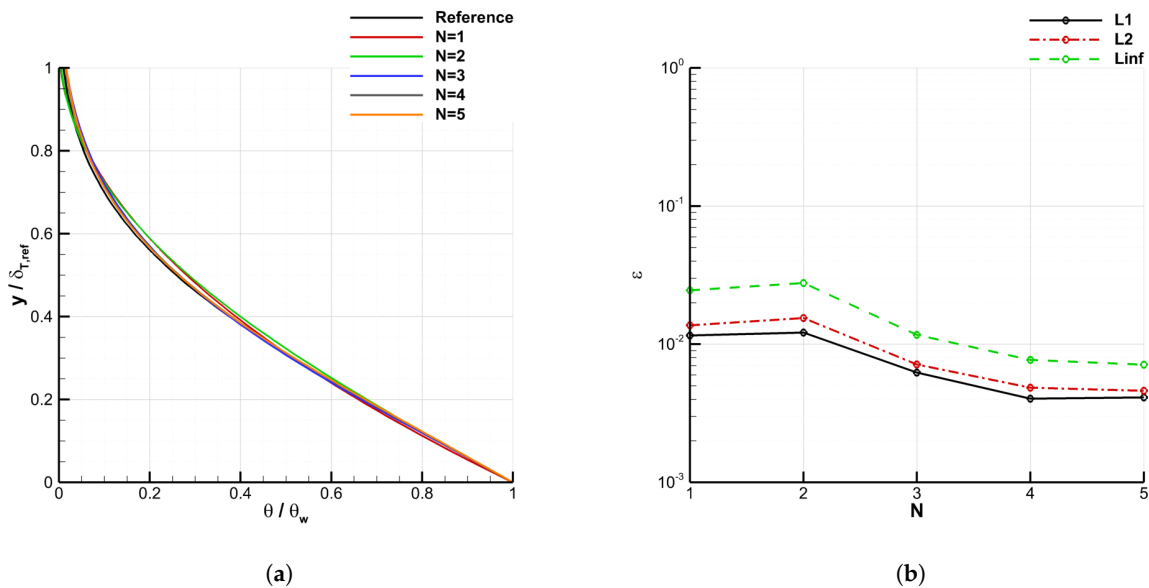


Figure 9. Temperature for $N = 1$ to $N = 5$, and for the reference CLICET solution. (a) temperature profiles at $x/L = 1$, (b) error. Test case 1, $N_x = 256$.

The main output expected from BLIM2D is the wall heat flux, which can be made non-dimensional via the Stanton number:

$$St = \frac{\phi_w}{\rho_e u_{xe} c_{pe} (-\theta_w)} \tag{51}$$

Figure 10a shows that the Stanton number is indeed little affected by N and the solution provided by BLIM2D is satisfactory. However, for this case, the Stanton number varies significantly because the heat flux ϕ_w is almost constant and the velocity increases linearly from $x = 0$ to $x = L$. The relative error between the results produced by BLIM2D and CLICET shows that the modal approach also produces the lowest errors for $N = 4$ and $N = 5$ (Figure 10b). There is no significant difference between the errors for $N = 4$ and $N = 5$, suggesting that the solution is converged for $N = 4$. The relative error is around 7×10^{-3} . Again, regarding the targeted applications, this can be deemed as very low.

The main explanation for this residual error is that for this case, as mentioned earlier, the source term G_i linked to the convection in the y direction cannot be neglected for Equations 2 to N (Equation (39)). Since the velocity fields are not exactly the same for CLICET and BLIM2D, the fact that G_i is not negligible is a cause of the residual error in ϕ_w : the use of the reference velocity field provided by CLICET in BLIM2D for u_y allows us to decrease the relative error in St to 4×10^{-3} . Additionally, an intrinsic source of error of the method is due to the definition used for the velocity profile u_y for $\delta < y \leq \delta_T$. The use of Equation (47) with $d_x(u_{xe}) \neq 0$ tends to arbitrarily extend u_y linearly for $y > \delta$. This does not make any real physical sense on the one hand, and on the other hand it introduces a difference with respect to CLICET, which has an evolution of u_y bounded between $y = 0$

and the reference boundary-layer thickness $\delta_{T,ref}$ (as shown in Figure 2). This ultimately disturbs the calculation of the integral (39) for this case where $\delta < \delta_T$.

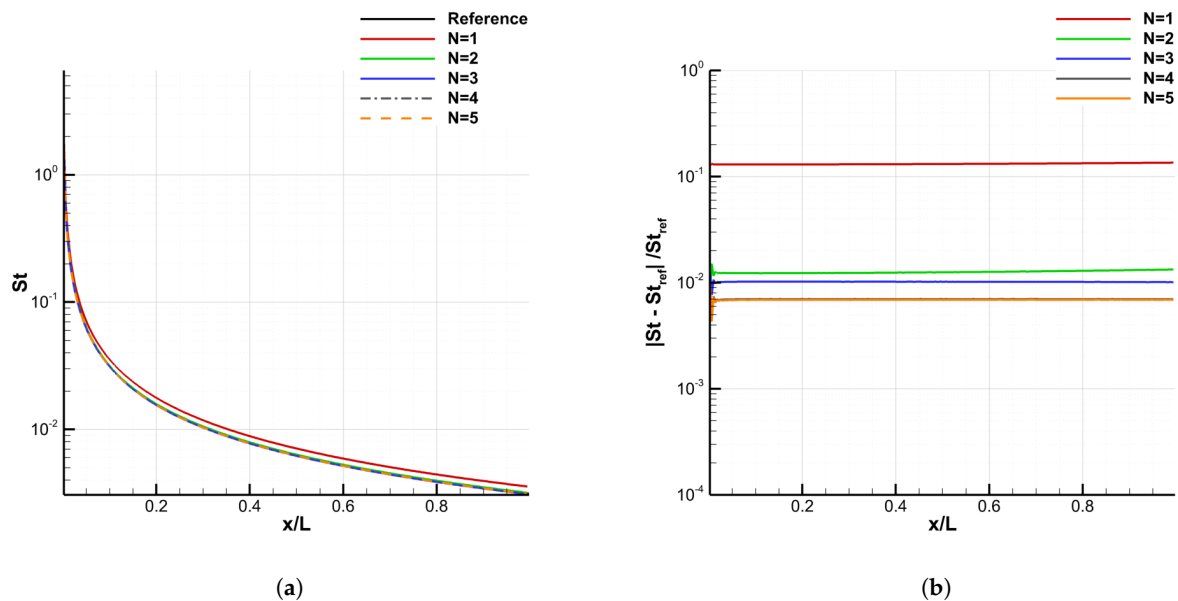


Figure 10. Stanton number for $N = 1$ to $N = 5$, and for the reference CLICET solution. (a) Stanton number, (b) relative error. Test case 1, $N_x = 256$.

4.3. Influence of the Wall to Free-Stream Temperature Ratio

In order to see the effect of the sole wall temperature on the solution (at fixed-edge temperature), the conditions of test case 2 were used (Table 1). These conditions are the same as test case 1, except that the edge-velocity exponent m can change and the wall temperature can be increased to values investigated in further test cases: $T_w \in \{273.15, 323.15, 373.15\}$ K (meaning that $T_w - T_e \in \{10, 60, 110\}$ K). The same grid was used as in Section 4.2.3, so $N_x = 256$.

4.3.1. Influence on the Velocity Profile

For $m = 1$, Figure 11a shows that the higher the wall temperature, the higher the discrepancy between the reference velocity profile and the profiles used in BLIM2D that do not account for any effect of the temperature on the density changes.

It has to be noted that $m = 1$ provides a better overall agreement between BLIM2D and CLICET on the velocity profiles (even up to $T_w - T_e = 110$ K) than any other value of pressure-gradient exponent m , as shown in Figure 11b. Less-accelerated ($m = 1/3$ or $m = 0$) or decelerated flows ($m = -0.01961$) exhibit larger L_2 relative errors in the velocity:

$$\begin{aligned} \varepsilon_{r,u_x,L2} &= \left(\frac{2}{L} \int_{L/2}^L \frac{1}{\delta_{ref}} \int_0^{\delta_{ref}} \left| \frac{u_x(x,y) - u_{xref}(x,y)}{u_{xref}(x,y)} \right|^2 dy dx \right)^{0.5}, \\ \varepsilon_{r,u_y,L2} &= \left(\frac{2}{L} \int_{L/2}^L \frac{1}{\delta_{ref}} \int_0^{\delta_{ref}} \left| \frac{u_y(x,y) - u_{yref}(x,y)}{u_{yref}(x,y)} \right|^2 dy dx \right)^{0.5} \end{aligned} \quad (52)$$

Additionally, the errors tend to increase when $T_w - T_e$ is increased (except incidentally the error in u_y for $T_w - T_e = 10$ K). Finally, the relative error is larger in u_y than in u_x . However, while the relative error increases when m decreases for u_x , there is no monotonic trend for the evolution of $\varepsilon_{r,u_y,L2}$ with respect to m .

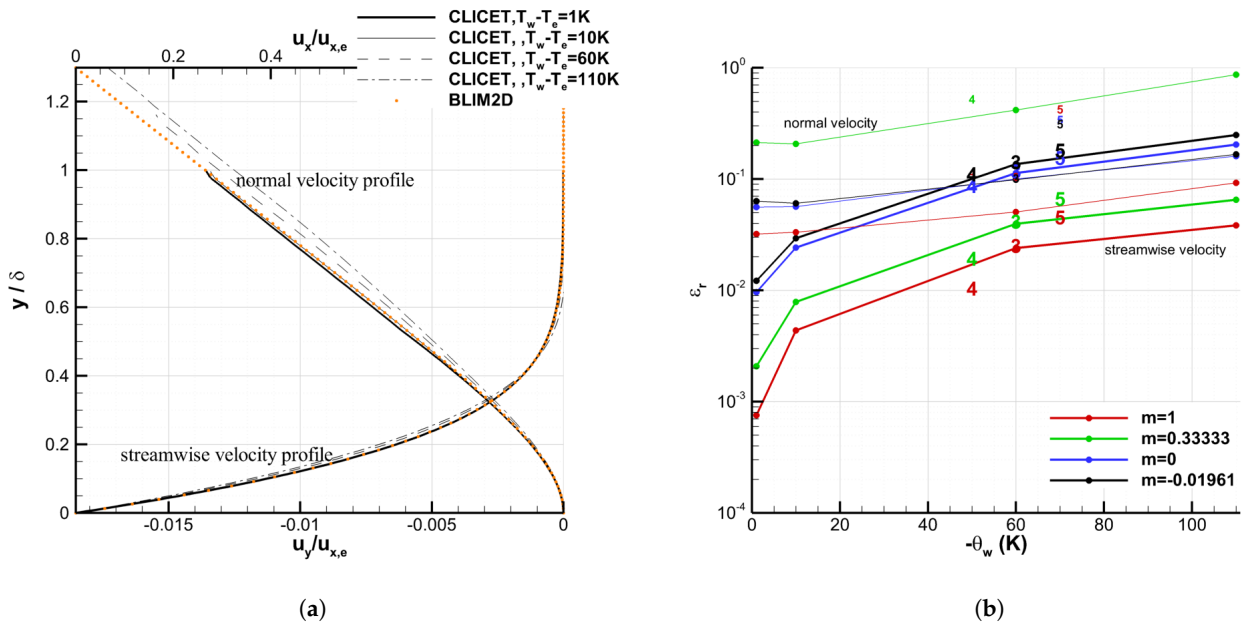


Figure 11. Velocity in test-case 2. (a) velocity profiles, $m = 1$, $x/L = 1$, (b) L_2 relative error. Impact of the wall temperature changes on the profiles. The errors of test cases 3, 4 and 5 are super-imposed (large case size: u_x ; small case size: u_y). $N_x = 256$.

4.3.2. Consequences on the Wall Heat Flux

The error in the wall heat-flux prediction is shown in Figure 12 (for $N = 4$). Except for the stagnation point case $m = 1$, the same trend is observed on the L_2 relative error

$$\epsilon_{r,\phi_w,L2} = \left(\frac{2}{L} \int_{L/2}^L \left| \frac{\phi_w(x) - \phi_{w,ref}(x)}{\phi_{w,ref}(x)} \right|^2 dx \right)^{0.5} \quad (53)$$

as for the velocity profiles, the error increases when $T_w - T_e$ is increased. $\epsilon_{r,\phi_w,L2}$ also tends to increase when m decreases. For the worst case, $m = -0.01961$ and $\theta_w = -110$ K, the relative error in the wall heat flux reaches 0.07. So, the relative error remains moderate and is related to the errors in the input velocity profiles. Notice that for $m = 1$, the relative error in the wall heat flux is even smaller: 7×10^{-3} for both $\theta_w = -1$ K and $\theta_w = -10$ K. However, it unexpectedly even drops lower than 10^{-3} for larger wall temperatures ($\theta_w = -60$ K and $\theta_w = -110$ K). This surprising trend is attributed to the fact that the source term G_i related to the convection in the y direction, which has an especially high contribution for $m = 1$. Since, additionally, $d_x(u_{x,e})$ has a higher value than for other values of m , as discussed earlier, the uncertainty is greater on the estimate of G_i than for the other values of m .

4.4. Influence of the Wall Temperature Gradient

Test cases more representative of flight conditions are addressed in this section. In-flight icing often occurs for moderate Mach numbers. Here, the value $M = 0.25$ was retained. Several conditions of wall temperature are investigated (cases 3 with constant temperature, 4 with negative wall-temperature gradient and 5 with positive wall-temperature gradient, see Table 1). Additionally, the number of grid elements is still $N_x = 256$ for all the simulations.

The relative errors in velocity profile of test cases 3, 4 and 5 are superimposed on the test case 2 curves in Figure 11b (test case numbers using a font color associated with the value of m). For the cases of non-zero gradient of temperature, the associated wall temperature is approximated, since it is not constant. The order of magnitude of the relative errors in the streamwise velocity is barely affected by the changed conditions. However, for the normal velocity, the error can be higher. In particular, the relative error can even

largely exceed 100% for some cases: (1) where u_y becomes low in a fairly extensive region near the wall ($m = 1/3$, $\vec{\nabla}T_w \cdot \vec{e}_x = 0$ K/m), and (2) where the sign of u_y changes along y ($m = 1/3$, $\vec{\nabla}T_w \cdot \vec{e}_x = 400$ K/m; $m = 0$ or -0.1961 , $\vec{\nabla}T_w \cdot \vec{e}_x = -400$ K/m).

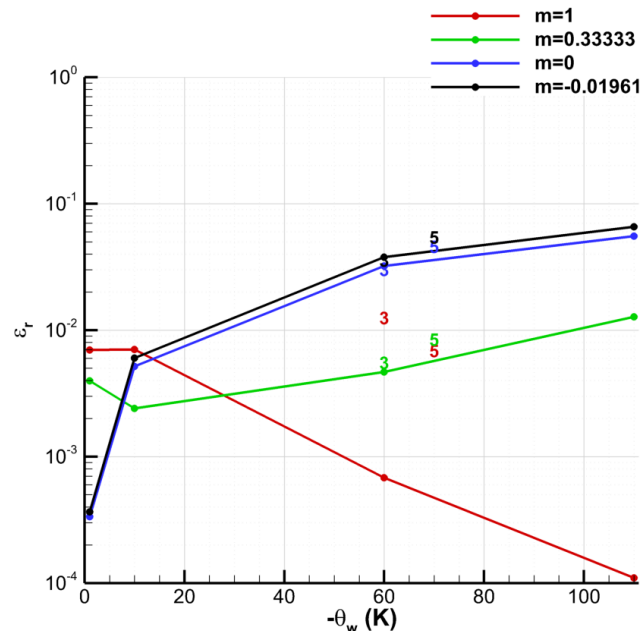


Figure 12. L_2 relative error in the wall heat flux in test-case 2 with respect to the wall temperature. The results of test cases 3, 4 and 5 are super-imposed. $N_x = 256$, $N = 4$.

Regarding the relative error in the wall heat flux, Figure 12 shows the errors for test cases 3, 4 and 5 in the same way as Figure 11b for the error in the velocity. The number of modes is $N = 4$ for all the solutions of this figure. The relative error exceeds 100% for test-case 4 because the heat flux changes sign, as shown later on. The relative error is thus very large in the area where the heat flux vanishes. Otherwise, the same orders of magnitude of error are retrieved as for the very low Mach number case 2. Moreover, the difficulties of the detailed analysis of the $m = 1$ case persist. The main conclusion is however that the relative error in heat flux does not exceed 7% for the cases investigated, which is mainly due to the errors in input-velocity profiles. This error is deemed reasonable for the envisioned applications.

In the remainder of this section, a more detailed analysis of the $m = 1$ and $m = -0.01961$ cases is made. Indeed, $m = 1$ is the stagnation-point condition, which is a key area for icing applications as already pointed out earlier. Moreover, the decelerated case $m = -0.01961$ exhibits the highest relative errors in heat flux.

4.4.1. Uniform Temperature, $m = 1$

First, the test-case 3 with a constant wall temperature is investigated and $N \in \{3, 4, 5\}$. Figure 13 shows the Stanton number evolution as well as the relative error compared to the reference solution. The solution for $N = 4$ achieves convergence in a number of modes, similarly to the low Mach number condition investigated earlier. The relative error increases along the profile, reaching 2% for $x/L = 1$. This is actually consistent with the fact that the error in the velocity was also observed to be increasing with x/L (which is not shown, for the sake of conciseness).

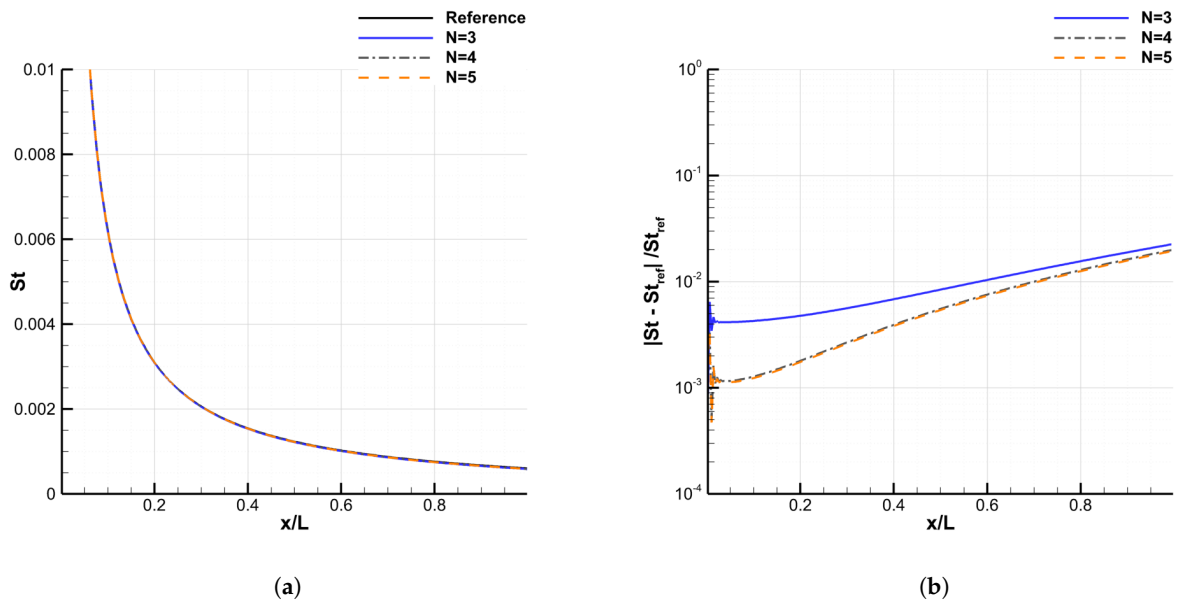


Figure 13. (a) Stanton number for $N = 3$ to $N = 5$ and (b) relative error in the Stanton number. Test case 3, $m = 1$. $N_x = 256$.

The solution is also good for the temperature profiles. The two temperature profiles shown in Figure 14 are very similar and the profiles produced by the various values of N are quite satisfactory.

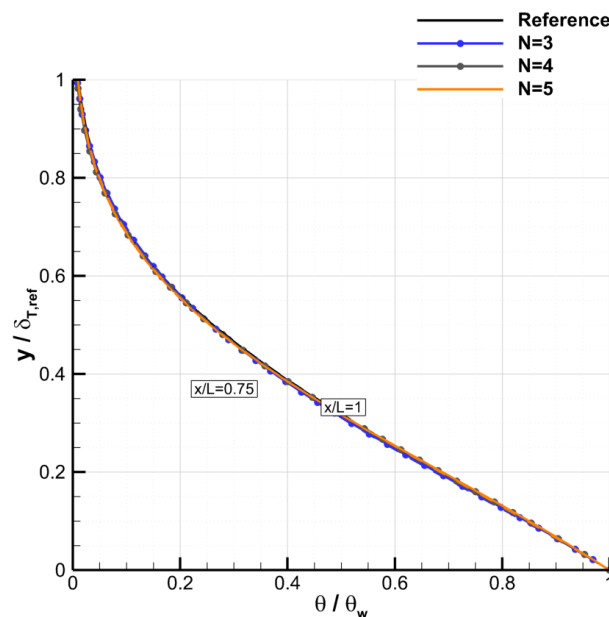


Figure 14. Temperature profiles for several x/L locations, and for the reference CLICET solution. Test case 3, $m = 1$. $N_x = 256$.

4.4.2. Uniform Temperature Gradient -400 K/m, $m = 1$

The negative wall temperature gradient test case is such that the wall temperature is 373.15 K for $x/L = 0$ and 273.15 K for $x/L = 1$. The wall heat flux is originally positive because the wall is warmer than the flow (and so is St , Figure 15a). Nevertheless, since the wall is colder further downstream, the airflow eventually carries enough heat for the wall heat flux to change sign, around $x_c/L = 0.75$ here. For $x/L > x_c/L$, the wall actually cools the airflow. The area in the vicinity of x_c/L is difficult to capture (the $N = 3$ curve

starts departing from the reference curve there). Increasing N is extremely beneficial for capturing the whole trend further downstream.

The temperature profiles shown in Figure 15b are indeed not always monotonic ($x/L = 0.875$ and $x/L = 1$ for instance). The heat transportation by the airflow actually heats the bottom part of the profile. This is more and more obvious for increasing values of x/L . This kind of profile is difficult to capture with temperature profiles in the form of Equation (17). Increasing the number of modes N allows higher-degree ($N + q$) polynomials for the modeling of the temperature profile. Convergence is thus achieved around $N = 10$ only and the trend of the reference profile is better captured with the highest values of N . This is also observed in $x/L = 0.75$ where the wall heat flux vanishes.

The error in the temperature profile increases at $x/L = 1$. This is partly due to the fact that the error in the vertical velocity profile is larger at this location than upstream. This effect is even more pronounced because at this location, the vertical convection source term G has a greater relative importance in the overall balance. Moreover, the need to locally refine the mesh to follow the increasing gradient of the solution is not totally excluded. Despite this weakness, the reference solution is very well reproduced up to $x/L = 0.875$ and the error on the Stanton number remains acceptable.

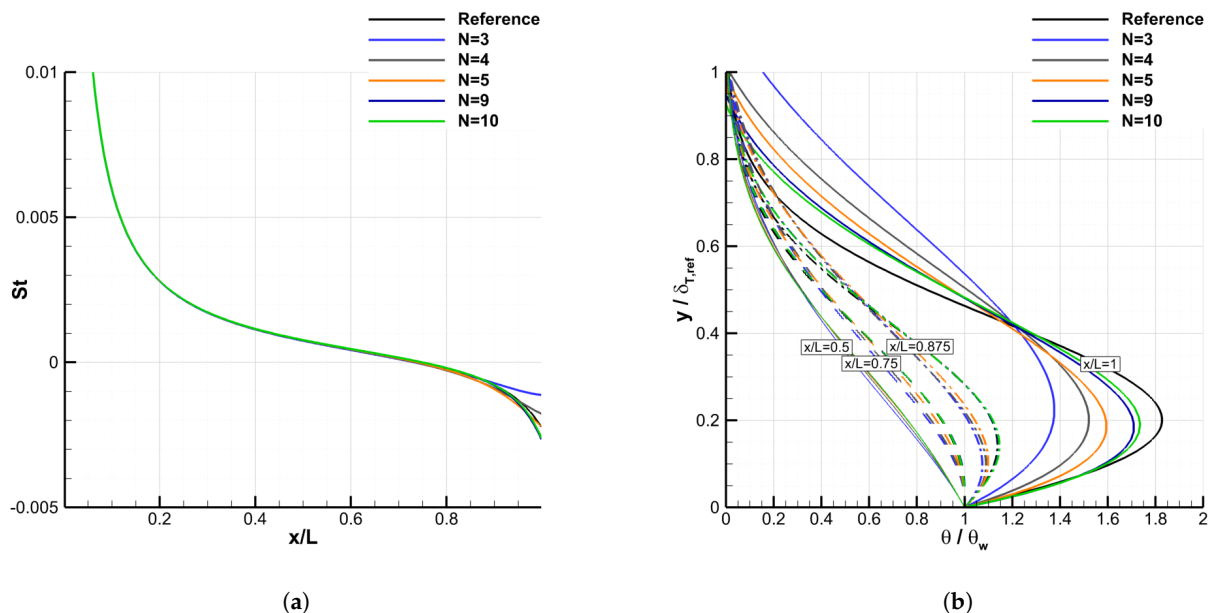


Figure 15. (a) Stanton number and (b) temperature profiles for several x/L locations. Test case 4, $m = 1$. $N = 3$ to $N = 10$, $N_x = 256$.

4.4.3. Uniform Temperature Gradient +400 K/m, $m = 1$

The positive wall temperature gradient test case is such that the wall temperature is 273.15 K for $x/L = 0$ and 373.15 K for $x/L = 1$. The wall thus keeps heating the airflow for increasing values of x/L . Figure 16 shows that this situation is less challenging for the modeling than negative wall temperature gradients. Despite the high level of wall temperature gradient addressed, the conclusions are quite similar to the zero-temperature gradient configuration. The solution is converged for $N = 4$. As shown in Figure 16a, the wall heat flux is accurately predicted (the relative error is actually lower than 2%). The temperature profiles are almost self-similar and they are accurately predicted by the Galerkin approach.

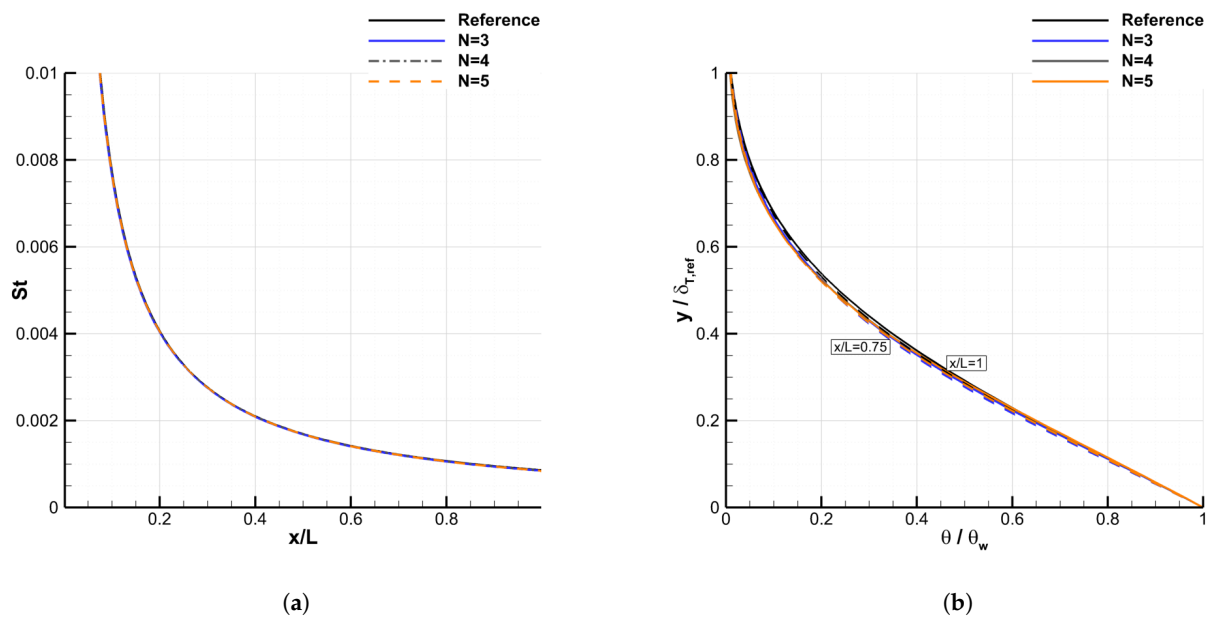


Figure 16. (a) Stanton number and (b) temperature profiles for several x/L locations. Test case 5, $m = 1$. $N = 3$ to $N = 5$, $N_x = 256$.

4.4.4. Uniform Temperature, $m = -0.01961$

Regarding the decelerated flow with zero wall temperature gradient, the conclusions are not very different from those obtained for $m = 1$, except that the error is overall larger. As shown in Figure 17, convergence is achieved for $N = 4$. The relative error in the Stanton number is large in the vicinity of $x/L = 0$ and approaches 4% for $x/L = 1$ (Figure 17a). As for $m = 1$, the slope of the error in St versus x/L is consistent with the slope observed for the error in the velocity (which also decreases with x). This confirms the major effect of the velocity profile provided as input to the calculation. Moreover, the catastrophic level of error for low x/L is actually deemed to be less representative of the behavior of the method than the final 4% error. A growing error in the dynamic boundary-layer resolution was indeed also observed in the vicinity of $x/L = 0$ for decelerated wedge flows in the paper of Bayeux et al. [12], but there was not such a strong error in the decelerated areas on real airfoils. Regarding the temperature profiles, self-similarity is almost retrieved and all values of N produce similar levels of error, which are higher than for $m = 1$ (Figure 17b).

As for $m = 1$, it has to be noticed that these conclusions remain valid for a heated wall with positive wall temperature gradient (test-case 5).

4.4.5. Uniform Temperature Gradient -400 K/m, $m = -0.01961$

Finally, the decelerated flow with negative wall temperature gradient is even more challenging for the model than for $m = 1$. The same conclusions can be drawn, but with higher levels of error, as shown in Figure 18. The number of modes required for achieving convergence is also larger ($N = 12$). The temperature profile is highly disturbed by the advection of heat from upstream to downstream (Figure 18b). Increasing the number of modes N allows us to better capture the temperature profiles, especially in the vicinity of the wall. As for $m = 1$, although the $N = 5$ solution is not fully converged and does not predict the reference temperature profile very accurately, the prediction of wall heat flux is already quite satisfactory, as shown in Figure 18a.

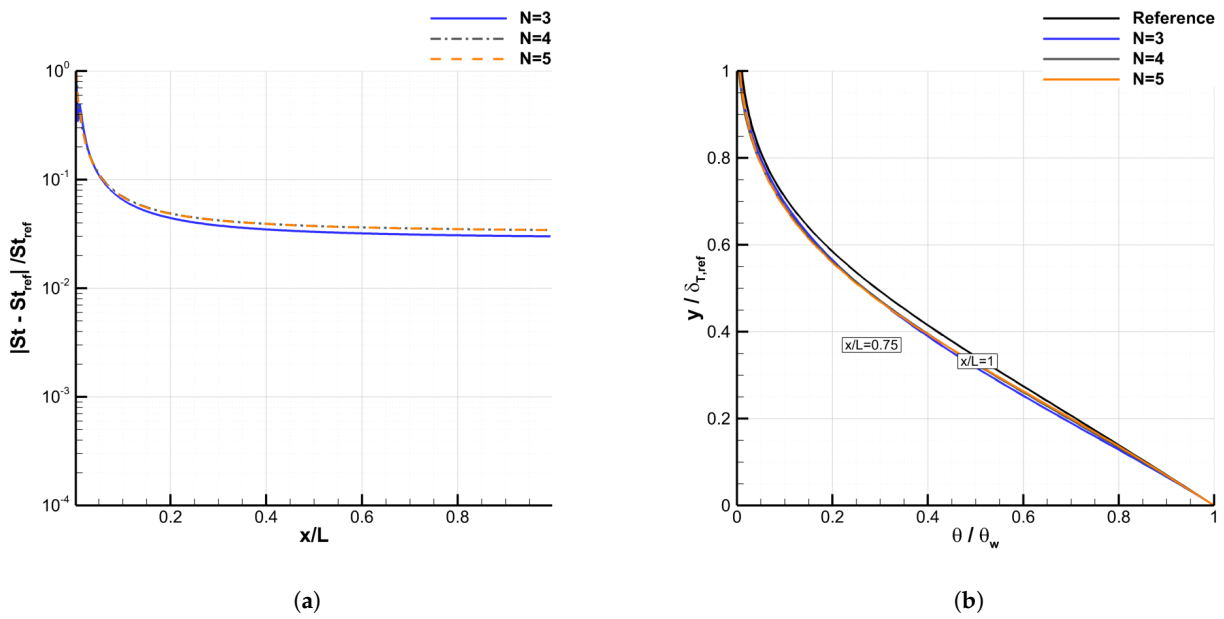


Figure 17. (a) Error in the Stanton number and (b) temperature profiles for several x/L locations. Test case 3, $m = -0.01961$. $N = 3$ to $N = 5$, $N_x = 256$.

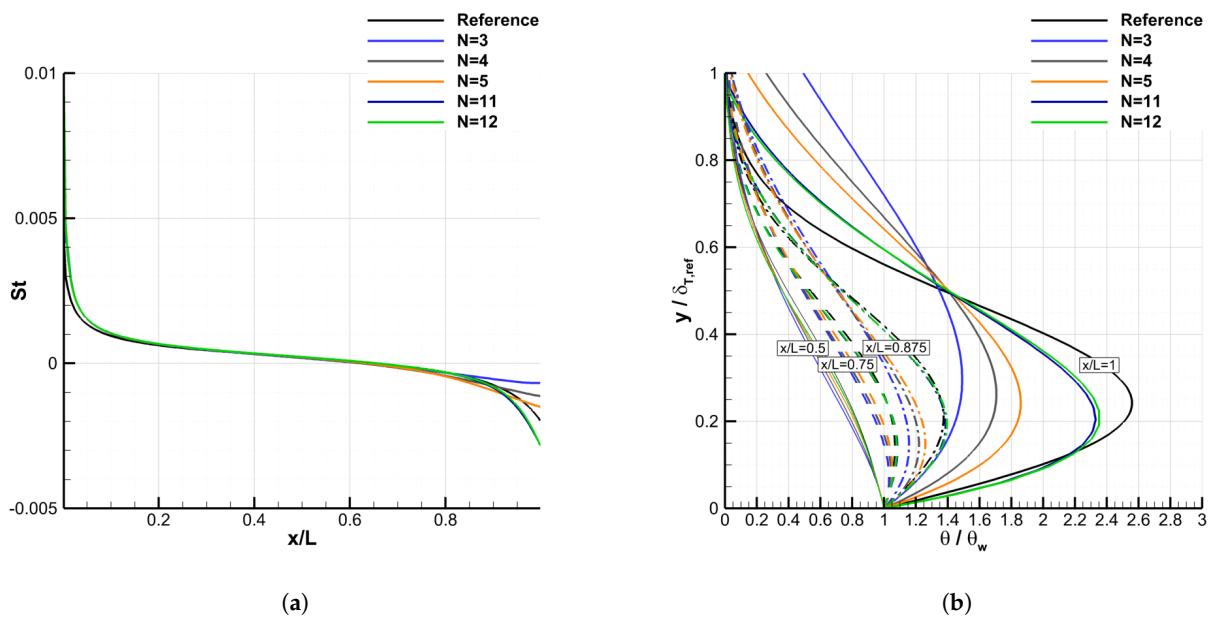


Figure 18. (a) Stanton number and (b) temperature profiles for several x/L locations. Test case 4, $m = -0.01961$. $N = 3$ to $N = 12$, $N_x = 256$.

4.5. Discussion about the Computational Time

It should be noted that the performance of the proposed method in terms of computational time has not been studied in this work. In particular, no effort has been made to optimize the code used. However, we can mention that the simulations performed with $N = 4$ to 5 modes, which are generally of sufficient accuracy, require only a few tens of seconds of computation on a simple single-core computer.

On the other hand, the computational time increases very quickly with the number N of modes because of the very restrictive stability condition (in $O(1/N^2)$) on the time step. The future use of a totally implicit time integration method, allowing us to get rid of the stability condition, should allow a gain of several orders of magnitude on the number of time steps necessary to reach the stationary solution and thus significantly increase the

performance of the method. Moreover, it should be kept in mind that for applications to aircraft in-flight icing, an important part of the expected time saving comes from the fact that the method should make it possible to replace iterative Navier–Stokes calculations by a viscous inviscid interaction approach, which is much less costly because it requires much less dense meshes near the walls.

5. Conclusions

In this paper, an original numerical method has been presented to solve laminar thermal boundary layers with non-uniform wall boundary conditions. This new method is based on the coupling between a modal method (Galerkin method) in the normal direction to the wall and a finite volume method in the tangential directions, which allows us to use only a surface mesh like an integral method.

Many numerical tests were performed to evaluate its accuracy, to determine the number of modes needed, and to ensure that the method is able to correctly compute the wall heat flux even in the presence of very large wall temperature variations. A series of academic test cases was used, corresponding to wedge flows, representative of the different possible situations encountered on a real airfoil protected by a thermal anti-icing or de-icing system: stagnation point flow, accelerated flow, and decelerated flow, with a positive or negative wall temperature gradient. The solutions obtained with the new method have been compared to those obtained with a finite volume method applied directly to the Prandtl equations of the boundary layer (ONERA's CLICET code). With only 5 modes, it has been shown that in all cases the relative error on the wall heat flux did not exceed a few percent, which is quite sufficient for the targeted applications considering the numerous other sources of uncertainty. It has also been shown that the origin of this error comes for a large part from the error committed on the prediction of the velocity field used for the resolution of the thermal boundary layer.

Future work will focus on the extension of the method to the case of turbulent boundary layers and on the implementation of the method in the ONERA 3D icing computational suite [26,27]. In particular the possibility of a two-way coupling with the resolution of the dynamic boundary layer will be considered and the explicit time integration scheme used in the present work will be replaced by an implicit scheme to avoid the restrictive stability condition and save computational time. Considering the good results obtained on the 2D laminar cases presented in this paper, this new method is a promising option to build a fast and robust computational suite to simulate the ice accretion on 3D surfaces protected by a thermal anti-icing or de-icing system.

Author Contributions: Conceptualization, P.V., R.H. and E.R.; methodology, P.V., R.H. and E.R.; software, R.H., P.V. and E.R.; validation, E.R.; formal analysis, E.R., P.V. and R.H.; investigation, E.R., P.V. and R.H.; data curation, E.R.; writing—original draft preparation, E.R. and R.H.; writing—review and editing, E.R. and P.V.; visualization, E.R.; supervision, P.V. and E.R.; project administration, E.R. and P.V.; funding acquisition, E.R. and P.V. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: The authors would like to thank G. Blanchard for the supportive and fruitful discussions.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Short Description of the Resolution of the Dynamic Boundary Layer in the Solver BLIM2D

In BLIM2D, the dynamic boundary layer is solved via a two-equation integral boundary-layer model:

$$\partial_t(\mathbf{U}) + \partial_x(\mathbf{F}(x, \mathbf{U})) = \mathbf{S}(x, \mathbf{U}) \tag{A1}$$

where

$$\mathbf{U} = \begin{pmatrix} u_{xe}\delta_1 \\ u_{xe}^2\delta_2 \end{pmatrix} \quad \mathbf{F} = \begin{pmatrix} u_{xe}^2\delta_2 \\ u_{xe}^3\delta_3 - u_{xe}^3\delta_2 \end{pmatrix} \quad \mathbf{S} = \begin{pmatrix} -u_{xe}\delta_1\partial_x(u_{xe}) + \frac{1}{2}u_{xe}^2C_f \\ (u_{xe}^2\delta_1 - u_{xe}^2\delta_2)\partial_x(u_{xe}) - \frac{1}{2}u_{xe}^3C_f + 2u_{xe}^3C_D \end{pmatrix}$$

δ_1 , δ_2 and δ_3 correspond to the displacement, momentum and kinetic-energy thicknesses, respectively.

$$\begin{aligned} \delta_1 &= \int_0^\infty \left(1 - \frac{u_x}{u_{xe}}\right) dy \\ \delta_2 &= \int_0^\infty \frac{u_x}{u_{xe}} \left(1 - \frac{u_x}{u_{xe}}\right) dy \\ \delta_3 &= \int_0^\infty \frac{u_x}{u_{xe}} \left(1 - \frac{u_x^2}{u_{xe}^2}\right) dy \end{aligned}$$

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

$$\frac{1}{2}u_e^2C_f = \frac{\tau_w}{\rho}, \quad u_e^3C_D = \frac{1}{\rho} \int_0^\infty \tau_{xy}\partial_y(u)dy$$

τ_w is the skin friction. In Equation (A1), the first equation is an unsteady version of the well-known von Kármán equation. The second one is the kinetic energy integral equation. In laminar regimes, this system is closed through the use of the following assumed velocity profile:

$$\frac{u_x}{u_{xe}} = 1 - (1 + a(H)\eta)(1 - \eta)^{p(H)-1} \tag{A2}$$

where $\eta = y/\delta$, δ is the dynamic boundary-layer thickness, and the parameters a and p are dependent on the shape factor $H = \frac{\delta_1}{\delta_2}$.

The conservative formulation allows the use of a finite volume resolution. A first-order upwind scheme is used and the steady-state solution is reached with a semi-implicit scheme, for which only the source terms are implicit, and with local time-stepping. The interested reader will find further information in Bayeux’s article [12].

Appendix B. Validation of the Model against an Empiric Relation

The method was assessed against the empiric relation $Nu = 0.453Re^{1/2}Pr^{1/3}$ valid for constant heat flux on a flat plate. The Reynolds number is $Re = \frac{\rho_e x u_{xe}}{\mu_e}$, whereas the Nusselt number is $Nu = \frac{hx}{k}$, where the heat transfer coefficient is $h = \frac{\phi_w}{T_w - T_r}$. The classic recovery temperature formula is used $T_r = T_e \left(1 + Pr^{1/2} \frac{\gamma - 1}{2} M^2\right)$. As stressed by Lienhard [28] for instance, it is expected that imposing a wall temperature varying as $x^{1/2}$ will lead to a constant wall heat flux along the flat plate. The conditions 3 of Table 1 were used with $m = 0$, except that the wall temperature is given by $T_w - T_r = 20x^{1/2}$. Regarding the mesh size, again, $N_x = 256$. Figure A1a shows that for a number of modes $N = 4$, the solution is well converged. The figure also demonstrates that the heat flux

is a rather well-predicted constant (relative error around 0.005), except in the immediate vicinity of $x = 0$. Figure A1b validates the model in terms of Nusselt number for this case with non-uniform wall temperature.

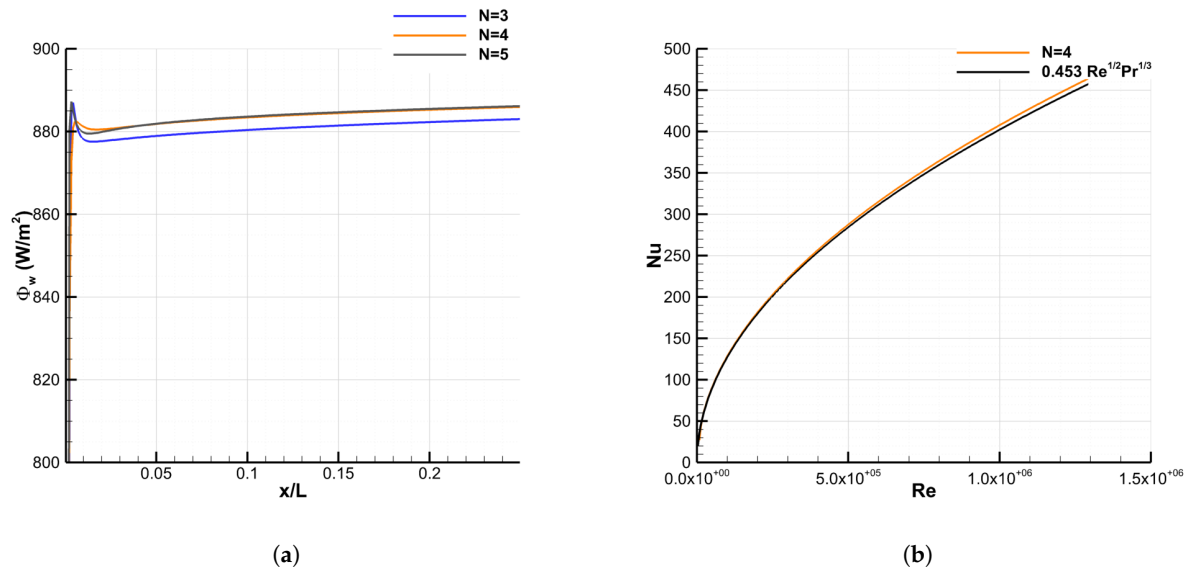


Figure A1. Validation of (a) the wall heat flux and (b) the Nusselt number. Test case 3, with wall temperature varying as $x^{1/2}$, $m = 0$. $N = 3$ to $N = 5$, $N_x = 256$.

References

- Morency, F.; Tezok, F.; Paraschivoiu, I. Anti-icing system simulation using CANICE. *J. Aircr.* **1999**, *36*, 999–1006. [\[CrossRef\]](#)
- Al-Khalil, K.M.; Horvath, C.; Miller, D.R.; Wright, W. *Validation of NASA Thermal Ice Protection Computer Codes Part 3: Validation of ANTICE*; Technical Report, NASA TM 2001-210907; Glenn Research Center: Cleveland, OH, USA, 2001.
- da Silva, G.A.L.; de Mattos Silveiras, O.; de Jesus Zerbini, E.J.G. Numerical Simulation of Airfoil Thermal Anti-Ice Operation. Part 1: Mathematical Modeling. *J. Aircr.* **2007**, *44*, 627–633. [\[CrossRef\]](#)
- Wright, W. *User's Manual for LEWICE Version 3.2*; Technical Report, NASA/CR-2008-214255; Glenn Research Center: Cleveland, OH, USA, 2008.
- Bu, X.; Lin, G.; Yu, J.; Yang, S.; Song, X. Numerical simulation of an airfoil electrothermal anti-icing system. *Proc. Inst. Mech. Eng. Part G J. Aerosp. Eng.* **2013**, *227*, 1608–1622. [\[CrossRef\]](#)
- Morency, F.; Tezok, F.; Paraschivoiu, I. Heat and Mass Transfer in the Case of Anti-Icing System Simulation. *J. Aircr.* **2000**, *37*, 245–252. [\[CrossRef\]](#)
- Croce, G.; Beaugendre, H.; Habashi, W. CHT3D-FENSAP-ICE conjugate heat transfer computations with droplet impingement and runback effects. In Proceedings of the 40th AIAA Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, 14–17 January 2002. [\[CrossRef\]](#)
- Bu, X.; Lin, G.; Shen, X.; Hu, Z.; Wen, D. Numerical simulation of aircraft thermal anti-icing system based on a tight-coupling method. *Int. J. Heat Mass Transf.* **2020**, *148*, 119061. [\[CrossRef\]](#)
- Henry, R. Development of an electrothermal de-icing/anti-icing model. In Proceedings of the 30th Aerospace Sciences Meeting and Exhibit, Reno, NV, USA, 6–9 January 1992. [\[CrossRef\]](#)
- Chauvin, R.; Villedieu, P.; Trontin, P.; Bennani, L. A robust coupling algorithm applied to thermal ice protection system unsteady modeling. In Proceedings of the 6th AIAA Atmospheric and Space Environments Conference, Atlanta, GA, USA, 16–20 June 2014. [\[CrossRef\]](#)
- Aupoix, B. *Couches Limites Bidimensionnelles Compressibles*; Descriptif et Mode d'emploi du Code CLICET—Version 2015; Technical Report; ONERA: Toulouse, France, 2015.
- Bayeux, C.; Radenac, E.; Villedieu, P. Theory and Validation of a 2D Finite-Volume Integral Boundary-Layer Method for Icing Applications. *AIAA J.* **2019**, *57*, 1092–1112. [\[CrossRef\]](#)
- Bayeux, C. Méthode Intégrale pour la Couche Limite Tridimensionnelle. Applications au Givrage. Ph.D. Thesis, Institut Supérieur de l'Aéronautique et de l'Espace (ISAE), Toulouse, France, 2017.
- Radenac, E.; Bayeux, C.; Villedieu, P. Use of a Two-Dimensional Finite Volume Integral Boundary-Layer Method for Ice-Accretion Calculations. *AIAA J.* **2020**, *58*, 1592–1606. [\[CrossRef\]](#)
- Radenac, E.; Kontogiannis, A.; Bayeux, C.; Villedieu, P. An extended rough-wall model for an integral boundary layer model intended for ice accretion calculations. In Proceedings of the 10th AIAA Atmospheric and Space Environments Conference—AVIATION 2018 ATLANTA, Atlanta, GA, USA, 25–29 June 2018. [\[CrossRef\]](#)

16. Harry, R.; Radenac, E.; Blanchard, G.; Villedieu, P. Heat Transfer modeling by integral boundary-layer methods towards icing applications. In Proceedings of the AIAA Aviation 2021 Forum, Virtual, 2–6 August 2021. [[CrossRef](#)]
17. Graebel, W.P.; Hsu, C. A Galerkin method for calculation of laminar thermal boundary layers. *Int. J. Heat Mass Transf.* **1981**, *24*, 771–774. [[CrossRef](#)]
18. Ozdemir, H.; Boogaard, E. Solving the integral boundary layer equations with a discontinuous Galerkin method. In Proceedings of the European Wind Energy Conference and Exhibition 2011, Brussels, Belgium, 13–17 March 2011.
19. Sunden, B.; Wu, Z. On Heat Transfer Issues for Wind Energy Systems. *J. Energy Resour. Technol.* **2017**, *139*, 051201. [[CrossRef](#)]
20. Lokatt, M.; Eller, D. Finite-volume scheme for the solution of integral boundary layer equations. *Comput. Fluids* **2016**, *132*, 62–71. [[CrossRef](#)]
21. Prieling, D.; Steiner, H. Analysis of the wall mass transfer on spinning disks using an integral boundary layer method. *Chem. Eng. Sci.* **2013**, *101*, 109–119. [[CrossRef](#)]
22. Cousteix, J. *Aérodynamique et Couche Limite Laminaire*; Cepadues Editions: Toulouse, France, 1989.
23. Papadakis, M.; Wong, S.H.; Yeong, H.W.; Wong, S.C. Icing Tunnel Experiments with a Hot Air Anti-Icing System. In Proceedings of the 46th AIAA Aerospace Science Meeting and Exhibit, Reno, NV, USA, 7–10 January 2008. [[CrossRef](#)]
24. Aupoix, B. Roughness corrections for the $k-\omega$ shear stress transport model: Status and proposals. *J. Fluids Eng.* **2015**, *137*, 021202. [[CrossRef](#)]
25. Chedevergne, F. Analytical wall function including roughness corrections. *Int. J. Heat Fluid Flow* **2018**, *73*, 258–269. [[CrossRef](#)]
26. Radenac, E.; Gaïble, H.; Bezar, H.; Reulet, P. *IGLOO3D Computations of the Ice Accretion on Swept-Wings of the SUNSET2 Database*; SAE Technical Paper 2019-01-1935; SAE: Warrendale, PA, USA, 2019. [[CrossRef](#)]
27. Bempedelis, N.; Bayeux, C.; Blanchard, G.; Radenac, E.; Villedieu, P. A 3D Finite-Volume Integral Boundary Layer method for icing applications. In Proceedings of the 9th AIAA Atmospheric and Space Environments Conference, Denver, CO, USA, 5–9 June 2017. [[CrossRef](#)]
28. Lienhard, J.H. Heat Transfer in Flat-Plate Boundary Layers: A Correlation for Laminar, Transitional, and Turbulent Flow. *J. Heat Transf.* **2020**, *142*, 061805. [[CrossRef](#)]