# Improved wall model treatment for aerodynamic flows in LBM

Johan Degrigny, Shang-Gui Cai, Jean-François Boussuge, Pierre Sagaut

▶ **To cite this version:**

Johan Degrigny, Shang-Gui Cai, Jean-François Boussuge, Pierre Sagaut. Improved wall model treatment for aerodynamic flows in LBM. Computers and Fluids, 2021, 227, pp.105041. 10.1016/j.compfluid.2021.105041 . hal-03597146

**HAL Id: hal-03597146**
**https://hal.science/hal-03597146**

Submitted on 4 Mar 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.

# Improved wall model treatment for aerodynamic flows in LBM

Johan Degrigny[a,b,*], Shang-Gui Cai[c], Jean-François Boussuge[a], Pierre Sagaut[c]

[a] CERFACS, 42 avenue Gaspard Coriolis, Toulouse 31057, France
[b] Airbus Operations SAS, 316, route de Bayonne Toulouse 31060, France
[c] Aix Marseille Univ, CNRS, Centrale Marseille, M2P2 UMR 7340, Marseille 13451, France

ARTICLE INFO

*Article history:*
Received 31 December 2020
Revised 5 May 2021
Accepted 2 June 2021
Available online 9 June 2021

*Keywords:*
Lattice Boltzmann method
Immersed boundaries
Wall model
Wall function
High Reynolds number
Reynolds-Averaged Navier-Stokes

ABSTRACT

The article deals with an improved treatment of wall models for the simulation of turbulent flows in the framework of Immersed Wall Boundaries on Cartesian grids. The emphasis is put on the implementation in a Lattice-Boltzmann Method solver without loss of generality, since the proposed approach can be used in Navier–Stokes-based solvers in a straightforward way. The proposed improved wall model implementation relies on the combination of several key elements, namely i) the removal of grid points too close to the solid surface and ii) an original computation of wall normal velocity gradient and iii) the interpolation scheme. The new method is successfully assessed considering URANS simulations focusing on steady solutions of the Zero Pressure Gradient turbulent flat plate boundary layer and the turbulent flow around a NACA0012 airfoil at several angles of attack.

## 1. Introduction

The growing interest for using LBM (Lattice Boltzmann Method) for high Reynolds number, moderate Mach number aerodynamic simulations in recent years can be explained by several factors. Firstly, this intrinsically unsteady method has a good computational efficiency for high fidelity scale-resolving computations like LES (Large Eddy Simulation). Simultaneously, the use of scale-resolving computations, fueled by expanding computational resources, enables capturing highly unsteady turbulent flows, such as massively separated flows, with a satisfactory accuracy. This remains a challenge for the industry standard RANS (Reynolds-Averaged Navier–Stokes) strategy. Secondly, the Cartesian grids used in LBM allow highly complex geometries to be considered, without the tedious and user-intensive generation of suitable body fitted meshes required by classic CFD solvers. Finally, the low numerical dissipation of standard LBM schemes is favorable for aeroacoustic applications, as well as for propagating wakes over long distances to investigate their effects on downstream elements.

Using wall models to avoid solving flows in the inner region of turbulent boundary layers at high Reynolds numbers is commonplace in LES [1,2]. Indeed, predictions [3,4] indicate that the cost of a wall-resolved LES computation with anisotropic grids in the range of Reynolds numbers of full scale aeronautical flows (around $10^7$ to $10^8$) would be one or more orders of magnitude larger than their wall-modelled LES (WM-LES) equivalents. This comes as a consequence of the increasing separation of the scales of the outer flow and the scale of the turbulent structures to be resolved in the inner boundary layer as the Reynolds number is increased.

In RANS computations, the wall-normal grid spacing requirement near the wall is essentially the same as in LES (around 1 wall unit, such that the first point is located in the viscous sublayer), for resolving the strong mean velocity gradients. The grid spacing in the wall-parallel direction, however, can be much looser, leading to very high aspect ratios of the grid near the wall. In combination with the absence of time-resolution, this greatly reduces the cost incentive for using wall models, typically algebraic ones referred to as wall functions [5].

In the context of Cartesian grids, the grid spacing in all directions is constrained by the most stringent requirement in any direction e.g. the wall-normal direction in boundary layers. The merely geometric implications of this matter, in comparison to the anisotropic grids, are examined for the part of the inner boundary layer extending from the wall to 50 wall units, covering the viscous sublayer and the buffer layer independently of the Reynolds number. Assuming a geometric growth rate of about 1.10 for the wall-normal spacing (doubling of the grid spacing in all directions every 7 cells on a Cartesian grid), the number of nodes required to discretize a cube of 50 wall units can be estimated to be roughly 400 times larger than the equivalent anisotropic LES grid (based on streamwise and transverse spacings of 40 and 20 wall units respectively) and nearly $5 \times 10^4$ times larger than the equivalent

* Corresponding author.
*E-mail address:* johan.degrigny@cerfacs.fr (J. Degrigny).

anisotropic RANS mesh (assuming streamwise and transverse spacings of 500 and 200 wall units respectively). These cost considerations are not taking the required time resolution into account, which would depend on the numerical scheme and the turbulence modeling strategy. Wall modeling is thus a crucial enabler for simulating high Reynolds number wall-bounded flows with Cartesian grids with any CFD solver, even more so than for LES on conventional anisotropic grids.

Another primary challenge for all methods relying on Cartesian grids is that these can almost never be fitted to the shape of interest for applying boundary conditions, in particular for solid walls. The immersed boundary method (IBM), first introduced by Peskin [6] to simulate the blood flow around a heart valve, enables the real shape of solid boundaries to be taken into account, even if they do not coincide with the computational mesh. Following the classification used in the review of Mittal and Iaccarino [7], this original formulation uses a continuous forcing approach, as the effect of the boundary is distributed over many nodes in the vicinity of the boundary. The sharper boundary representation obtained with the discrete forcing approach [8,9], where the forcing due to the boundary is applied locally on the nodes next to it, is more appropriate for high Reynolds number boundary layers with their large gradients. This sparked the development of combinations of turbulent wall modeling with the IBM [10–12], although it has also been demonstrated with continuous forcing [13]. Unlike with body-fitted meshes, imposing the wall shear stress yielded by a wall model is not straightforward with the IBM. Even when the velocity is set to the appropriate value near the wall, special care may need to be taken for the shear stress to be consistent, in accordance with the numerical scheme used [14,15].

The diversity of the approaches proposed for implementing these boundary conditions in LBM highlights the difficulty of doing so accurately. In the frame of wet-node boundary conditions, some authors implement a Dirichlet velocity condition on near-wall nodes, either by completely reconstructing the distribution functions at the boundary [16,17] or by using an extrapolation scheme [18]. Maeyama et al. [19] impose the velocity on the first layer of nodes inside the solid body in the spirit of ghost-cell IBM, after a linearization procedure of the averaged near-wall velocity profile [20,21], also using an extrapolation scheme.

With link-wise boundary conditions, a local slip velocity can be defined at the wall, using the wall model to account for the unresolved part of the boundary layer. Nishimura et al. [22] used a linear extrapolation of velocity previously used with other IBM implementations [23,24] with the interpolated bounce back of Yu et al. [25], while Pasquali et al. [26] used Hoffman's approximation [27] with the interpolated bounce-back of Geier et al. [28]. The velocity can also be imposed on an imaginary surface inside the fluid (but close to the wall) [29] with the Filippova and Hänel boundary condition [30].

The possibility of prescribing an arbitrary wall shear stress condition [31] has also been exploited to directly enforce the wall shear stress from the wall model [32,33]. Within a volumetric LBM scheme, this strategy has been assessed for many aerospace applications. Aerodynamic validation cases on generic aircraft geometries have been published for transonic conditions [34] and realistic high lift configurations [35]. Various aeroacoustic cases have also been presented on airframe noise at full scale [36], turbofan noise [37], jet noise [38], and trailing edge noise [39].

A hybrid between both families has been proposed [40] by applying a velocity correction on boundary nodes after performing Bouzidi's interpolated bounce-back [41]. Most of these treatments were demonstrated in conjunction with the LES turbulence modeling strategy (WM-LES) [16,18,22,26,33,40], others with RANS models (WM-RANS) [17,29,32], and even with a combination of both [19] (WM-RANS-LES).

The RANS turbulence modeling approach is used exclusively in this study. Since the flow is solved time-accurately in LBM, even when coupled with a RANS model, Unsteady-RANS (URANS) computations are actually performed. For attached flows devoid of large scale unsteadiness or unsteady external forcing, steady solutions are nevertheless obtained after the initial transient phase. For the development of a wall treatment for high Reynolds number attached flows, which is the aim of this study, these steady solutions are easier to analyze and understand. URANS computations are not, however, the ultimately targeted application, as using an explicit time marching LBM solver for obtaining a steady RANS solution is inefficient with respect to dedicated finite volume RANS solvers. Additionally, using an isotropic grid provides little or no advantage over the usual highly stretched ones with RANS turbulence modeling and results in unnecessarily large grid sizes as outlined above. Nevertheless in the perspective of DES-type hybrid RANS-LES methods (see [42] for a review), it allows working specifically on the modeling for attached boundary layers which are meant to be captured in RANS mode.

This paper is organized as follows: after a general description of the numerical method, the novel wall model treatment is introduced. This treatment is then assessed on test cases of increasing complexity: a flat plate and a modified NACA0012 airfoil.

## 2. Numerical method

This study is based on an experimental branch of the commercial LBM solver ProLB (previously known as LaBS), developed within several scientific entities including CSSI, Renault, Airbus, École Centrale de Lyon, CNRS, and Aix-Marseille University, in partnership with CERFACS. Its formulation is node-based.

### 2.1. Lattice boltzmann method

The LBM is based on the statistical mechanics of particles at mesoscopic scale, i.e. a scale larger than the molecular scale but smaller than the macroscopic continuum scale. The method's working quantity is the probability distribution function $f(\boldsymbol{\xi}_i, \boldsymbol{x}, t)$ of the particles with a discrete velocity $\boldsymbol{\xi}_i$ at location $\boldsymbol{x}$ and time $t$. For convenience, the distribution function in the direction of $\boldsymbol{\xi}_i$ is denoted by $f_i$. The macroscopic properties of the fluid can be obtained by taking the statistical moments of the particle distributions

$$\rho(\boldsymbol{x}, t) = \sum_i f_i(\boldsymbol{x}, t), \tag{1}$$

$$\rho \boldsymbol{u}(\boldsymbol{x}, t) = \sum_i \boldsymbol{c}_i f_i(\boldsymbol{x}, t), \tag{2}$$

where $\boldsymbol{c}_i = \boldsymbol{\xi}_i/\sqrt{3}$ in the present case.

In the standard LBM, the particle distribution functions are discretized in space on a regular cubic lattice, where particles travel with their discrete velocity from one lattice site to another in one time step. The choice of the lattice is essential for the isotropy of the scheme and its physical properties. In ProLB, the classic D3Q19 (with 19 discrete velocities in 3 dimensions) lattice is used, which enables modeling weakly compressible isothermal flow. As illustrated in Fig. 1, the nodes are organized in cubic fashion and each node is connected to its direct neighbors by discrete velocities along the principal axes and the cubic edge centres.

The dynamics of the particle distributions on the lattice are given by the Lattice Boltzmann Equation (LBE)

$$f_i(\boldsymbol{x} + \boldsymbol{c}_i \Delta t, t + \Delta t) = f_i(\boldsymbol{x}, t) + \Omega_i(\boldsymbol{x}, t), \tag{3}$$

where $\Delta t$ is the time step. $\Omega_i(\boldsymbol{x}, t)$ is the collision operator, and it is the key to the physical modeling, as it models the interactions

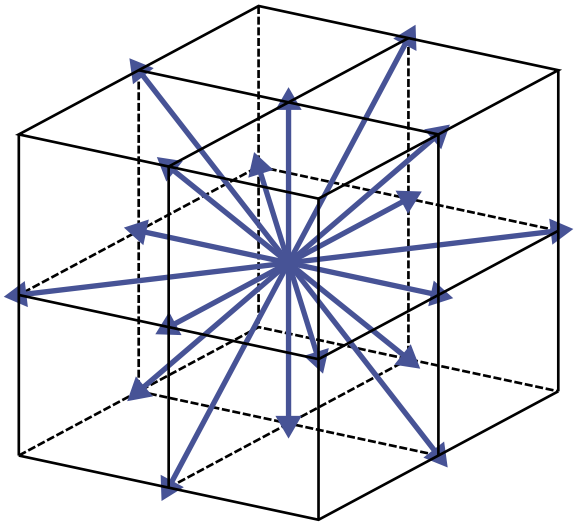

**Fig. 1.** Schematic of the D3Q19 lattice, with 19 velocities (null velocity not shown) in 3 dimensions.

of particles present at a node at a given time. The simple BGK collision operator [43] reads

$$\Omega_i = -\frac{1}{\tau}\left(f_i - f_i^{\mathrm{eq}}\right), \tag{4}$$

which corresponds to the relaxation of the particle distributions towards their equilibrium state $f_i^{\mathrm{eq}}$ in a characteristic relaxation time $\tau$. The discrete equilibrium functions $f_i^{\mathrm{eq}}$ are obtained from the macroscopic velocity $u_\alpha$ and density $\rho$ of the fluid with a discretized form of the Maxwell-Boltzmann thermodynamic equilibrium

$$f_i^{\mathrm{eq}} = w_i \rho \left(1 + \frac{c_{i\alpha}u_\alpha}{c_s^2} + \frac{u_\alpha u_\beta\left(c_{i\alpha}c_{i\beta} - c_s^2\delta_{\alpha\beta}\right)}{2c_s^4}\right), \tag{5}$$

with $c_s = 1/\sqrt{3}$ the speed of sound of the lattice and $w_i$ the weights for each discrete velocity, which are fixed for a given lattice. The deviation from this equilibrium is noted

$$f_i^{\mathrm{neq}} = f_i - f_i^{\mathrm{eq}}. \tag{6}$$

In the present study, the Hybrid Recursive Regularised BGK (HRR-BGK) collison model [44] is employed, which greatly improves the stability of the original BGK model at high Reynolds numbers.

The LBE can be explicitly advanced in time, through the "collide and stream" algorithm, alternating the instantaneous exchange of momentum of particles at each lattice node in the collision step, and the streaming of particle distributions along each velocity in time and space. Since the particle distributions travel directly form one lattice site to another without any loss or approximation, this scheme has a low numerical dissipation. The locality of the nonlinear collision operator and the simplicity of the streaming procedure confer it its computational efficiency and a high potential for parallelization.

The Chapman-Enskog expansion can be used to show that the macroscopic equations solved with the LBM in the hydrodynamic limit are actually the weakly compressible Navier–Stokes equations

$$\partial_t \rho + \partial_\gamma(\rho u_\gamma) = 0, \tag{7}$$

$$\partial_t(\rho u_\beta) + \partial_\alpha(\rho u_\alpha u_\beta) = -\partial_\beta p + \partial_\alpha\left(\rho\nu\left(\partial_\alpha u_\beta + \partial_\beta u_\alpha\right)\right), \tag{8}$$

with a cubic velocity deviation in the momentum equation which can be neglected in low Mach number flows. The macroscopic pressure and viscosity are recovered respectively by

$$p = c_s^2\rho, \tag{9}$$

$$\nu = c_s^2(\tau - \frac{1}{2}). \tag{10}$$

For a more thorough description of LBM, the reader is referred to [44,45].

### 2.2. Spalart–Allmaras turbulence model

For turbulence modeling, the $f_{v3}$ variant of the Spalart–Allmaras (S–A) turbulence model without the $f_{t2}$ term is employed in this study. It solves a single transport equation for the pseudo turbulent viscosity $\tilde{\nu}$, which can be rewritten as a function of the dimensionless viscosity ratio $\chi = \tilde{\nu}/\nu$ as follows:

$$\frac{\partial\chi}{\partial t} + u_j\frac{\partial\chi}{\partial x_j} = c_{b1}\tilde{S}\chi - c_{w1}f_w\nu\left(\frac{\chi}{d}\right)^2 + \frac{\nu}{\sigma}\left[\frac{\partial}{\partial x_j}\left((1+\chi)\frac{\partial\chi}{\partial x_j}\right) + c_{b2}\frac{\partial\chi}{\partial x_i}\frac{\partial\chi}{\partial x_i}\right], \tag{11}$$

where $\tilde{S}$ is defined by

$$\tilde{S} = f_{v3}\Omega + \frac{\chi\nu}{\kappa^2 d^2}f_{v2}, \tag{12}$$

with

$$f_{v2} = \frac{1}{(1+\chi/c_{v2})^3}, \qquad f_{v3} = \frac{(1+\chi f_{v1})(1-f_{v2})}{\chi},$$

$$\Omega = \sqrt{2\Omega_{\alpha\beta}\Omega_{\alpha\beta}}, \qquad \Omega_{\alpha\beta} = \frac{1}{2}\left(\frac{\partial u_\alpha}{\partial x_\beta} - \frac{\partial u_\beta}{\partial x_\alpha}\right),$$

$$c_{v2} = 5.$$

Compared to the original S–A model [46], the introduction of the $f_{v3}$ term eliminates the possibility of negative values of $\tilde{S}$ in the production source term. All other formulae and constants are the same as in the standard S–A model:

$$f_w = g\left[\frac{1+c_{w3}^6}{g^6+c_{w3}^6}\right]^{1/6}, \quad g = r + c_{w2}(r^6 - r) \quad r = \min\left[\frac{\chi\nu}{\tilde{S}\kappa^2 d^2}, 10\right],$$

$$c_{b1} = 0.1355, \qquad \sigma = \frac{2}{3}, \qquad c_{b2} = 0.622,$$

$$\kappa = 0.41, \qquad c_{w1} = \frac{c_{b1}}{\kappa^2} + \frac{1+c_{b2}}{\sigma}, \quad c_{w2} = 0.3, \qquad c_{w3} = 2.$$

The turbulent viscosity ratio is computed by

$$\frac{\nu_t}{\nu} = \chi f_{v1}, \tag{13}$$

with

$$f_{v1} = \chi^3/(\chi^3 + c_{v1}^3) \qquad \text{and} \qquad c_{v1} = 7.1,$$

which is added to the fluid's molecular viscosity and finally takes effect through the relaxation time in the LBM collision operator.

The S–A model transport equation is implemented in the LBM solver using finite differences on the Cartesian grid. The centered finite difference discretization is utilized for the derivatives, except for the convective term for which a hybrid centered/upwind scheme is applied. The free-stream value of $\chi = 3.0$ is imposed at the far field boundaries of the computational domain as recommended [46,47] while $\chi = 0$ holds at resolved walls.

### 2.3. Model consistent wall function

The common use of two-layer wall models in the context of the IBM [10,13,15,48,49], where the simplified turbulent boundary layer equations are solved on a secondary grid near the wall, once

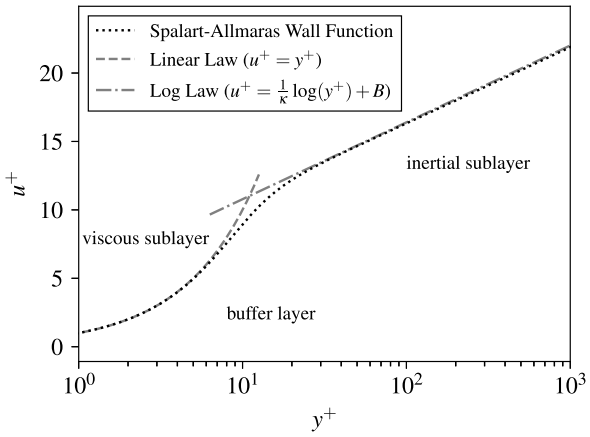


**Fig. 2.** Spalart–Allmaras wall function [47] velocity profile, compared to the linear law for the viscous sublayer and the logarithmic law of the wall (with $\kappa = 0.41$ and $B = 5.165$) for the inertial sublayer.

again emphasizes the importance of wall modeling for the computation of high Reynolds number boundary layers with the IBM. For simplicity, however, an algebraic wall model, i.e. a wall function, is chosen for modeling the under-resolved inner part of the boundary layer, since the improved wall treatment is the focus of this study, rather than advanced physical modeling.

Common relationships are readily available for the velocity in the viscous sublayer (linear law) and in the inertial sublayer of equilibrium boundary layers (logarithmic law of the wall), but they do not apply for the buffer layer between them. Even though several formulas have been proposed to provide a continuous representation of the inner boundary layer [50–52], their behaviour in the buffer layer is not necessarily in agreement with the profiles that would be obtained using a wall-resolved grid on a flat plate without pressure gradients with any particular RANS model. On Cartesian grids, it cannot easily be prevented that some near-wall nodes are located in the buffer layer. To avoid introducing errors due to this discrepancy, model-consistent wall functions (also called adaptive wall functions) have been used with the IBM. Bernardini et al. [53] used the look-up table technique to interpolate the result of a wall-resolved flat plate computation [54] to perform RANS and DDES computations with the S–A model. Zhou [24] used a blend of existing wall function profiles [55] in conjunction with the k-$\omega$ SST RANS model. In the present study, the analytical solution of the S–A model for a zero pressure gradient flat plate [47] is employed:

$$u^+(y^+) = \bar{B} + c_1 \log[(y^+ + a_1)^2 + b_1^2] - c_2 \log[(y^+ + a_2)^2 + b_2^2] - c_3 \operatorname{Arctan}[y^+ + a_1, b_1] - c_4 \operatorname{Arctan}[y^+ + a_2, b_2], \tag{14}$$

with $y^+ = \frac{y u_\tau}{\nu}$, $u^+ = \frac{u}{u_\tau}$, $u_\tau = \sqrt{\frac{\tau_w}{\rho}}$, $\tau_w$ the wall shear stress, and Arctan the mathematical equivalent of the FORTRAN function $atan2(x, y)$. The constants are:

$$\bar{B} = 5.0333908790505579, \quad a_1 = 8.148221580024245,$$
$$a_2 = -6.9287093849022945,$$
$$b_1 = 7.4600876082527945, \quad b_2 = 7.468145790401841,$$
$$c_1 = 2.5496773539754747,$$
$$c_2 = 1.3301651588535228, \quad c_3 = 3.599459109332379,$$
$$c_4 = 3.6397531868684494.$$

As shown in Fig. 2, the velocity profile described by this function approaches the linear law inside the viscous sublayer ($y^+ \lessapprox 5$) and the logarithmic law in the inertial sublayer ($y^+ \gtrapprox 40$). Barring numerical error [54], mesh convergence beyond the inertial range should be possible, at least on a flat plate without pressure gradients. It should be noted that this profile is obtained under the equilibrium assumption, where the pressure gradient and convection effects are neglected. Its accuracy is thus expected to degrade with significant deviations from these assumptions e.g. when the boundary layer is not fully developed, or in the presence of adverse pressure gradients.

An off-wall boundary condition also needs to be provided for the turbulent variable of the RANS model. Through its construction, the S–A model provides a simple linear relationship for the modified eddy viscosity valid for the whole inner boundary layer (including the buffer layer):

$$\chi = \kappa y^+. \tag{15}$$

## 3. Wall treatment

This section details the treatment of the wall boundary nodes, which are defined as nodes where the streaming operation cannot be performed due to the vicinity of the wall. It is based on the wall modeling approach, but since the wall shear stress cannot be imposed directly as in classic solvers, the tangential velocity (and its wall-normal gradient) are instead imposed on the on the boundary nodes. The particle distributions are then reconstructed from the macroscopic quantities to provide boundary conditions for the LBM scheme.

The present work does not consider the case in which the grid is not uniform along the wall —i.e. grid resolution interfaces intersect the wall— to focus on the wall model implementation without additional complications. In the case of a node-based solver with an overlap of the coarse and fine domains (co-located coarse and fine nodes at the interface) as described by Astoul et al. [58], the present method can be extended to these cases by simply treating each boundary node separately.

### 3.1. Near-wall interpolation

To apply the tangential velocity boundary condition, the wall function has to be inverted to determine the wall friction velocity $u_\tau$, based on the tangential velocity at a given distance from the wall. This can be done iteratively through a Newton solver when this relationship is implicit, as in Eq. 14. In some other wall treatments, the velocity is first interpolated at an artificial point (often referred to as image point, or reference point) at an arbitrary wall distance, and the wall function is inverted based on this interpolated velocity and the chosen wall distance [17,59]. Accurately interpolating the highly non-linear velocity profile in turbulent boundary layers is very challenging, and inaccurate interpolations, such as the IDW (Inverse Distance Weighing) used in [17], can lead to poor boundary layer modeling and oscillations in near-wall quantities [60]. Contrary to the fluid velocity, $u_\tau$ typically varies smoothly and can thus be more easily interpolated accurately, even using simple methods [60].

In the present wall treatment, a fixed stencil is used for selecting donor nodes, which includes both direct neighbors and some nodes from the second layer of nodes around the current boundary node, as illustrated in Fig. 3 (a.). Stencil locations inside the solid, or that match boundary nodes, are discarded so that only fluid nodes (which have already been computed at the current time step) are used as input. In a first step, $u_\tau$ is computed for each of these donor nodes by inverting the wall function with their respective tangential velocity and wall distance. In a second step, the $u_\tau$ at the current boundary node is estimated by averaging the values computed at the donor nodes, weighed by their inverse tan-

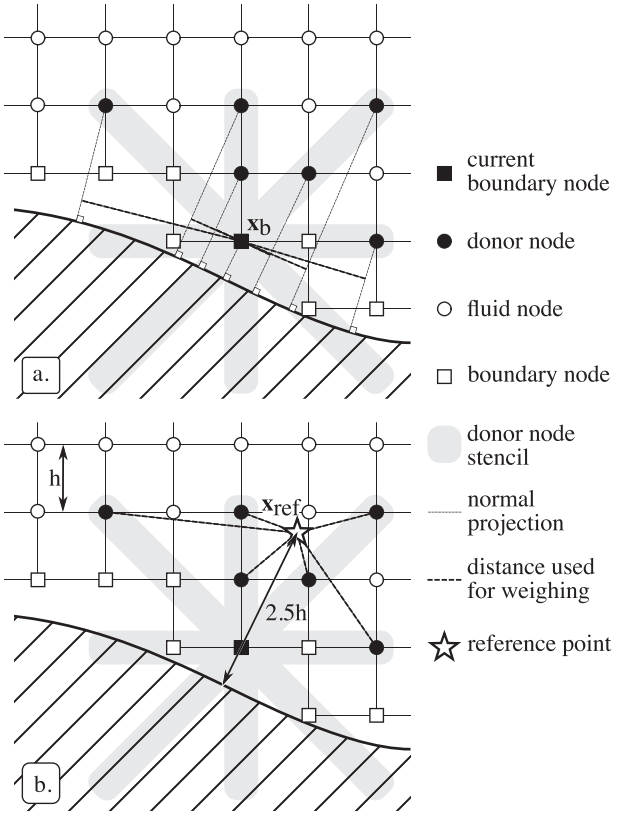


**Fig. 3.** Schematic of the interpolation of $u_\tau$ (a.) and of the simple IDW interpolation for the density (b.).

gential distance from it:

$$u_{\tau,\mathrm{b}} = \frac{\sum_i u_{\tau,i} a_i}{\sum_i a_i} \quad \text{with} \quad a_i = \frac{1}{\|(\mathbf{x}_i - \mathbf{x}_\mathrm{b}) - (\mathbf{x}_i - \mathbf{x}_\mathrm{b}) \cdot \mathbf{n}_i\|^2}, \tag{16}$$

where index $i$ loops over the available donor nodes, $\mathbf{x}_\mathrm{b}$ is the position of the current boundary node, $\mathbf{x}_i$ is the position of donor node $i$, and $\mathbf{n}_i$ is the unit normal vector to the wall at donor node $i$. Finally, the tangential velocity at the current boundary node is computed by evaluating the wall function with the spatially averaged $u_\tau$ and its wall distance.

The additional cost incurred by inverting the wall function at each donor node is not expected to be very significant, despite the increase of the number of times applying this procedure. Indeed, as the value of $u_\tau$ from the previous time step can be used as an initial guess for the Newton solver, the number of iterations performed each time should be minimal. This cost could be further reduced by using an explicit wall function [17,60–62].

There is no obstacle for employing the IDW interpolation at a reference point for the density. Indeed, if the pressure gradients in the wall normal direction inside the boundary layers are neglected, and since the scheme is isothermal, the density can be considered constant in the wall-normal direction. Furthermore, the density can also be expected to vary smoothly along the wall in subsonic flows. For setting the density at the current boundary node, a reference point is thus defined at a wall distance of $2.5h$ (where $h$ is the local grid spacing), on the wall-normal line passing through the boundary node as illustrated in Fig. 3 (b.). The density is then interpolated by simple IDW at this location from the same donor node cloud as for $u_\tau$:

$$\rho_\mathrm{ref} = \frac{\sum_i \rho_i b_i}{\sum_i b_i} \quad \text{with} \quad b_i = \frac{1}{\|\mathbf{x}_i - \mathbf{x}_\mathrm{ref}\|^2}, \tag{17}$$

where $\mathbf{x}_\mathrm{ref}$ is the position of the reference point. Neglecting wall-normal variations, the density at the current boundary node is set to be equal to that interpolated at the reference point.

### 3.2. LBM boundary condition implementation

To set up boundary conditions for LBM at the boundary nodes, the computed macroscopic quantities need to be translated into the corresponding particle distribution functions. For the sake of simplicity and robustness, a full reconstruction strategy is adopted for boundary nodes: all particle distributions are reconstructed, and the ones reaching boundary nodes through streaming are discarded. As discussed in Section 2.1, the particle distribution functions can be split into the equilibrium component $f^\mathrm{eq}$ that can be determined by the local fluid density and velocity though Eq. 5, and the non-equilibrium component $f^\mathrm{neq}$.The latter can be obtained in several ways: Haussmann et al. [18] extrapolated it from the inside of the fluid following Guo's scheme [63], Haussmann et al. [40] used an interpolated bounce-back operation [41], Malaspinas and Sagaut [16] exploited the symmetry properties of flat grid- aligned boundaries. The importance of the $f^\mathrm{neq}$ component to the accuracy of the wall treatment was evidenced by Haussmann et al. [18].

$f^\mathrm{neq}$ can also be approximately reconstructed from macroscopic quantities [56,57], and this stragety is followed here. Indeed the Chapman-Enskog expansion allows making a link between this non-equilibrium component and the strain rate tensor $\mathbf{S} = \left(\nabla\mathbf{u} + (\nabla\mathbf{u})^T\right)/2$ :

$$f_i^\mathrm{neq} \approx -\frac{w_i \tau}{c_s^2} \mathbf{Q}_i : \rho\mathbf{S}, \tag{18}$$

with $\mathbf{Q}_i = \boldsymbol{\xi}_i\boldsymbol{\xi}_i - c_s^2 I$.

In order to complete the reconstruction process, the strain rate thus has to be known or computed at boundary nodes. The difficulty of the numerical estimation of near-wall gradients on under-resolved grids for the implementation of boundary conditions in LBM has previously been noted [64].

In attached boundary layers, the wall-normal gradient of the tangential velocity is expected to be the largest contributor to the strain rate tensor, thus its estimation is of major importance for an accurate implementation of the wall function boundary condition. As the wall function describes the assumed shape of the inner boundary layer velocity profile though an explicit and differentiable formula (Eq. 14), an exact formula for the assumed wall-normal gradient can be obtained. Nevertheless, a discretized derivative is preferred in the present wall treatment to help account for the fact that the gradients are not exactly represented in the numerical scheme. To this end, the wall function is evaluated at the wall distance of the current boundary node $n_b^+$ and multiples of the local grid spacing $h^+$ (both in wall units) farther away. The derivative in the wall-normal direction ($n$) is thus computed through a third order one-sided finite difference

$$\left(\frac{\mathrm{d}u}{\mathrm{d}n}\right)_b = \frac{u_\tau}{h}\left(-\frac{11}{6}u^+\left(n_b^+\right) + 3u^+\left(n_b^+ + h^+\right) - \frac{3}{2}u^+\left(n_b^+ + 2h^+\right) + \frac{1}{3}u^+\left(n_b^+ + 3h^+\right)\right). \tag{19}$$

This process is completely local (once $u_\tau$ is determined) since the wall function is simply evaluated at arbitrary wall distances i.e. no additional data from the flow field is used. Preliminary studies have shown that using the exact formula degrades the results with respect to the discretized one.

This discretized wall-normal gradient is then transformed into the grid axes for entering the particle distribution reconstruction process while the other components of $\nabla\mathbf{u}$ are computed by fi-

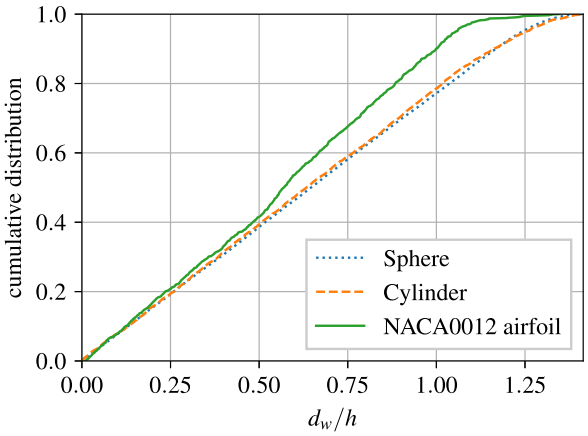


**Fig. 4.** Examples of cumulative distributions of the wall distance of boundary nodes on a sphere, a cylinder, and a NACA0012 airfoil, each with uniform grid spacing at the surface.

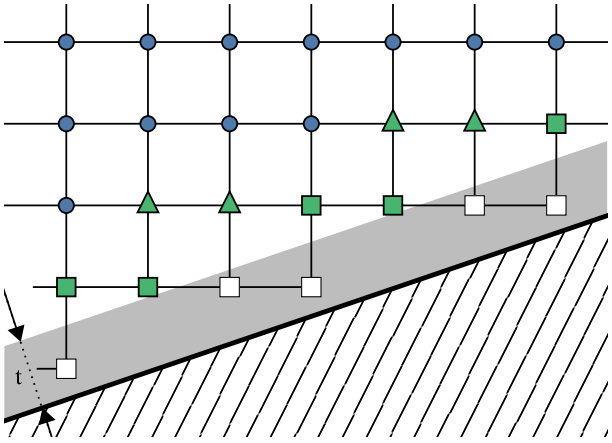


**Fig. 5.** Schematic of the elimination of boundary nodes located closer to the wall than the threshold value $t$. White squares: excluded original boundary nodes; green squares: unaffected original boundary nodes; green triangles: original fluid nodes that effectively become boundary nodes; blue circles: unaffected fluid nodes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

nite differences, namely second order centered where geometrically possible and first order one-sided otherwise.

Since the wall function only provides information on the tangential velocity, a distinct method is needed for its normal component. In the present wall treatment, the normal velocity is set to 0 at the boundary nodes i.e. the non-penetration boundary condition valid at the wall is applied directly on the boundary nodes. This first order approximation reinforces numerical robustness and is not expected to have a large impact on the overall precision of the computation since the normal component is minor. This full reconstruction process does not *a priori* conserve mass at the wall boundary, but it can potentially be combined with a correction for mass conservation such as the one of Bocquet et al. [65] if required.

### *3.3. Elimination of nodes too close to the wall*

The staircase patterns of the wall boundaries generated around arbitrary geometries introduce geometrical irregularities, even on smooth surfaces. Indeed the wall distances $d_w$ of the boundary nodes are spread over a large range from 0 to $\sqrt{2}h$ (the longest lattice link) and their distributions cannot be prescribed for the generation of grids around realistic geometries. Several examples are shown in Fig. 4 for a sphere, a cylinder, and the NACA0012 aifroil. In these examples, a large number of boundary nodes are located very close to the wall, with wall distances much smaller than the local cell size. For instance about 20% are within $0.25h$ from the wall, and the rate at which nodes located infinitesimally close to the wall occur –represented by the slope of the cumulative distribution at the origin– is not lower than that of being farther away.

For these nodes located much closer to the wall than the local grid spacing, the wall model becomes essentially inoperative. Indeed the wall model is used to take into account the large near-wall gradients that cannot be resolved by the numerical scheme for a given grid. Hence if a node is located too close to the wall with respect to the grid spacing (i.e. where the gradients cannot be resolved), an accurate solution cannot be computed. This situation could also be interpreted as a locally unbounded growth rate of the grid spacing in the wall normal direction, as the wall distance of boundary nodes approaches 0. Furthermore, the discontinuities occurring at the steps introduce large ratios of the wall distance between adjacent nodes along the wall. This will cause large velocity ratios along the wall which can be an additional challenge to the numerical scheme for yielding smooth and accurate solutions.

In the present wall treatment, a threshold $t$ is defined on the wall distance of boundary nodes, in the local grid spacing unit, to designate boundary nodes deemed too close to the wall. Boundary nodes which wall distance is smaller than this threshold ($d_w < th$) are completely excluded from the computation to avoid the previously mentioned issues. To accommodate this change, the wall boundary condition has to be applied on some fluid nodes that have lattice links to the excluded boundary nodes, with which the streaming of distributions can no longer take place. They essentially become boundary nodes following the application of the threshold, as illustrated in Fig. 5. The excluded boundary nodes can still be computed through the wall treatment for cosmetic purposes, they can be given arbitrary values, or they can even be removed from the data structure if it is convenient to do so, e.g. to reduce the memory requirements. If they remain in memory, they have to be excluded from all other computations inside the solver, in particular for computing gradients. The threshold value should be chosen between 0 and 1, as there would be no use in excluding boundary nodes that are already farther away from the wall than the local grid spacing. In the following, the expression "boundary nodes" refers to the original boundary nodes that are located beyond the threshold —thus retaining their function— as well as the fluid nodes behaving as boundary nodes.

This technique effectively shifts the application of the wall boundary condition away from the walls where the wall distance is deemed too small, whereas it remains unchanged where the wall distance is deemed large enough. The range of the wall distance distribution of boundary nodes becomes $th$ to $(t+\sqrt{2})h$ as the entire distribution is shifted away from the wall, as illustrated in Fig. 6 on a NACA0012 airfoil. Since the wall distance can be kept on the order of the grid spacing, the boundary nodes can potentially be placed in the inertial sublayer through careful grid generation, for the optimal application of algebraic wall functions.

The effective wall location and shape in the computation remain unchanged by this procedure, as the true wall distance and wall normal direction remain used for all boundary nodes. Since the location at which the wall boundary condition is imposed is merely shifted in space, the computational cost of this treatment is not expected to be significantly affected by the threshold value. A slight increase in the number of boundary nodes can be expected on convex surfaces and conversely a slight decrease on concave ones.

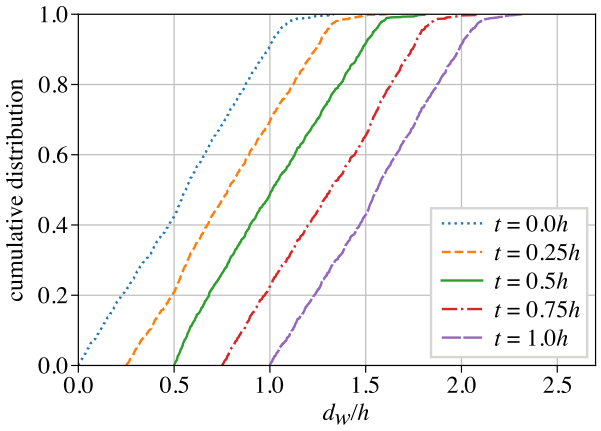


**Fig. 6.** Cumulative distribution of the wall distance of boundary nodes on a NACA0012 airfoil with uniform grid spacing at the surface, for different threshold values $t$ for the elimination of nodes too close to the wall.

Another approach to remove issues due to boundary nodes located too close to walls would be to implement the wall treatment on the second layer of nodes away from the wall, thereby excluding all original boundary nodes. This measure is deemed excessive, as a non negligible part of these natural boundary nodes (around 22% for the example of the sphere in Fig. 4) are already located farther from the wall than the local grid spacing. These nodes are not part of the problem, and there is no reason to exclude them. Indeed if this alternative was employed, the range of wall distance of effective boundary nodes would be extended from $h$ to $2\sqrt{2}h$. This larger range would make it more difficult to locate all nodes within the same region of the boundary layer, and the grid spacing would have to be reduced by about 27% in comparison the threshold method to maintain the nodes farthest from the wall within the same range of wall distance.

This technique can be related to the projection technique used to construct the near-wall cells in the simplified cut-cell method of Harada et al. [59], whereby the nodes closest to the walls are eliminated. The application of the velocity boundary condition on an arbitrary off-wall surface of Filippova et al. [29] would likely provide similar benefits if it is located far enough from the wall. Vis-à-vis the proposed threshold technique, these alternatives are likely to eliminate more near-wall nodes than necessary.

### 3.4. Summary of the wall treatment

The wall treatment procedure in the present LBM solver is summarized below. It is applied on all effective boundary nodes, once they have been identified after applying the wall distance threshold.

1. Select the fluid nodes inside a fixed stencil as donor nodes
2. Compute $u_\tau$ for each donor node by inverting the wall function (Eq. 14) with its tangential velocity and wall distance
3. Interpolate $u_\tau$ from the donor nodes in the wall tangential direction (Eq. 16).
4. Interpolate the density at the reference point from the donor nodes by simple IDW (Eq. 17) and set the same density to the current boundary node.
5. Evaluate the wall function (Eq. 14) at the current boundary node with the interpolated $u_\tau$ to obtain the tangential velocity and set the normal velocity to 0.
6. Compute the velocity gradients by finite differences and rectify the normal gradient component with the discretized derivative of the wall function profile (Eq. 19)

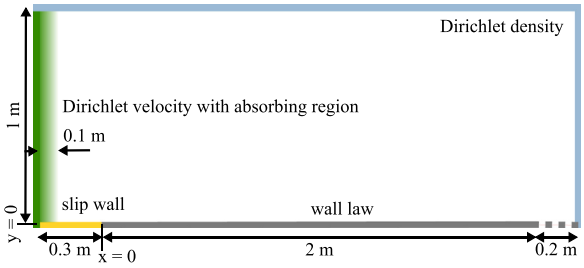


**Fig. 7.** Schematic of the computational set up for the flat plate case.

7. Reconstruct the equilibrium $f^{\text{eq}}$ (Eq. 5) and the non-equilibrium $f^{\text{neq}}$ (Eq. 18) components of the particle distribution function from the velocity, density, and the velocity gradient.

The baseline treatment is very similar to that described by Wilhelm et al. [17], except with the model consistent wall function used in this study. It is identical to the present one but with the 3 main improvements removed. Instead of interpolating the $u_\tau$ computed at each donor node, the tangential velocity at the arbitrary reference point is interpolated from that at the donor nodes by simple IDW (as described for the density) and the wall function is inverted based on this interpolated velocity; instead of using the discretized wall function gradient, all velocity gradients used for the reconstruction of particle distributions at the current boundary node are computed by finite differences using the surrounding nodes; instead of eliminating boundary nodes too close to the wall, all boundary nodes are computed normally.

## 4. Validation

### 4.1. Zero pressure gradient flat plate

#### 4.1.1. Test case presentation

In the context of the IBM, the flat plate offers the opportunity of validating boundary layer modeling on a body-fitted grid, when the plate is aligned with the grid axes. Although this simple academic case is not very representative of real applications it does allow testing the wall treatment in the absence of an irregular grid topology at the boundaries. The aim of this test is to evaluate the accuracy of the boundary layer modeling using the present wall treatment and its ability to reach grid convergence in ideal conditions where the wall function is fully consistent with the expected solution and the grid is body fitted.

The wall treatment is applied on a 2 m long plate (the reference length is taken as the half-length $L = 1$ m). The freestream Mach number is $\text{Ma} = 0.2$ and the Reynolds number based on the reference length is $\text{Re}_L = 5 \times 10^6$. This validation case is given in NASA's Turbulence Modeling Resource [66] although some of its specifications were adapted for the present study. As illustrated in the schematic in Fig. 7, a constant velocity is imposed on the inflow boundary, while a constant pressure is imposed on the outflow and upper boundaries of the domain. As an unsteady compressible solver is used with reflective boundary conditions, an absorbing region (following [67,68]) is placed next to the inlet boundary to allow acoustic perturbations to be damped inside the domain. A short section of slip wall upstream of the plate and an extension of the plate downstream are excluded from the analysis, and aim at avoiding any interference between the wall treatment and the other boundary conditions. As true 2D computations are not possible with the ProLB solver, a thin slice of volume is computed, with periodicity conditions applied on the front and back faces. The minimum width of the slice is constrained by the coarsest grid spacing, since the grid is isotropic. The width of the domain is thus

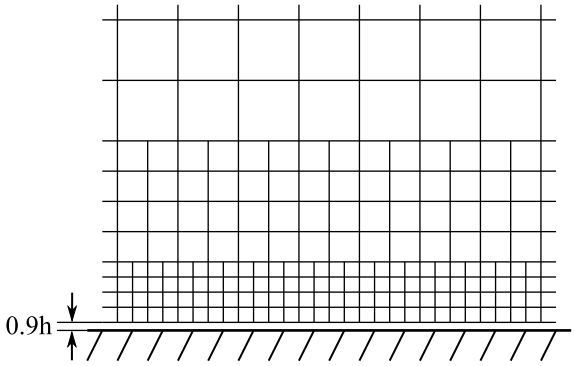


**Fig. 8.** Schematic of the mesh construction including the wall distance of the first nodes.

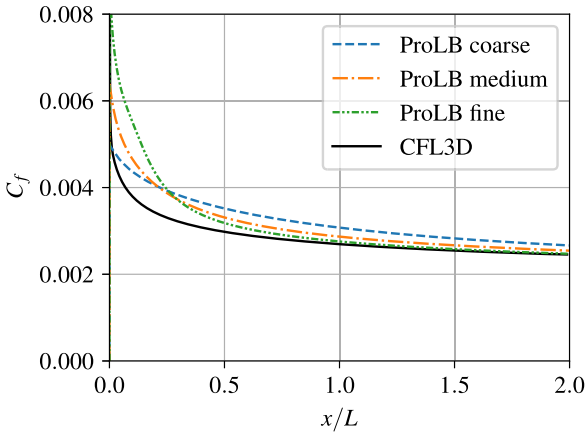


**Fig. 9.** Skin friction coefficient along the flat plat plate for different grid spacings, compared to CFL3D results from NASA's Turbulence Modeling Resource [66].

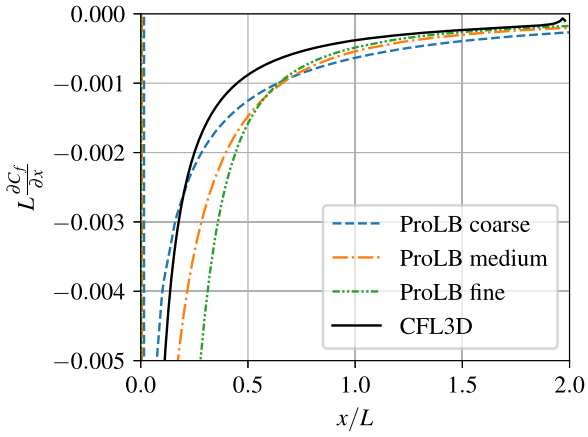


**Fig. 10.** Longitudinal gradient of the skin friction coefficient along the flat plat plate for different grid spacings, compared to CFL3D results from NASA's Turbulence Modeling Resource [66].

equal to the grid spacing away from the plate, which is 8 times larger than the spacing at the plate.

The present wall function based wall treatment is significantly different from the resolved adiabatic solid wall condition specified in [66] and differences are expected to appear, especially close to the upstream end of the plate. Indeed the wall function cannot perform accurately where the boundary layer is not thick enough for the donor nodes to be located within the range of validity of the wall function (viscous sublayer, buffer layer, and inertial layer, as illustrated in Fig. 2). These errors will inevitably be convected downstream and affect the boundary layer state on the whole plate. Nonetheless, this configuration is representative of real aerodynamic flows, where boundary layers develop from zero thickness, and this error should be accounted for in the overall accuracy of the wall model. It is thus justified to compare the results obtained with the proposed wall treatment to computational results obtained with a classic RANS solver on a wall-resolved mesh, which are representative of the results expected with the S–A turbulence model on this configuration.

The grid spacing is kept uniform along the wall to avoid refinement interfaces cutting through the boundary layer. The grid is thus constructed in streamwise uniform layers as illustrated in Fig. 8, with the grid spacing doubling at each interface. Each layer comprises approximately 40 cells in the wall-normal direction to reduce the influence of refinement interfaces parallel to the plate. As this grid construction allows choosing a value for the wall distance along the plate, which is uniform, a wall distance of $0.9h$ is chosen. As the wall distance could be explicitly controlled, the elimination of boundary nodes too close to the wall is not needed and is thus deactivated.

#### *4.1.2. Results*

To begin, the behaviour of the skin friction coefficient along the plate, defined as:

$$C_f = \frac{\tau_w}{1/2\rho_\infty V_\infty^2}, \tag{20}$$

is examined for coarse, medium, and fine grids with near wall spacings of $h/L = 7.5 \times 10^{-4}$, $h/L = 2.5 \times 10^{-4}$, and $h/L = 6.25 \times 10^{-5}$ respectively. On each grid, the accuracy of the skin friction improves as the boundary layer develops along the plate, as shown in Fig. 9, in comparison with the reference computation performed with CFL3D [66]. Downstream of $x/L \approx 0.25$, the more refined grids yield more accurate skin frictions, with a good agreement between the fine grid result and the reference on the second half of the plate.

The upstream extremity of the plate constitutes a singularity, as the skin friction should be infinite at this point, which is challenging for any solver. The skin friction at this location increases with the grid refinement yet its very steep decrease along the plate is not well captured. Indeed, the donor nodes of the first boundary nodes at the beginning of the plate are located outside the boundary layer, which is just starting its spatial development in that area, such that the value of $C_f$ there is based solely on the freestream velocity and the wall distance (proportional to the grid spacing). As the boundary layer needs to develop for the wall function modeling to be accurate, an adaptation distance can be observed before the physical modeling reaches its full potential accuracy.

The gradient of the skin friction along the plate, shown in Fig. 10, confirms that between $x/L = 0.25$ and $x/L = 0.5$, the $C_f$ decreases faster as the grid is refined. Towards the downstream end of the plate, the skin friction gradient approaches that of the reference as the grid is refined. In the reference solution, a perturbation is visible near $x/L = 2$ due to the outflow boundary at the end of the plate, which is avoided in the present computations by extending the plate a little farther.

To further examine the accuracy of the boundary layer modeling, the velocity profile is extracted for each computation at the location where the Reynolds number based of the momentum thickness $Re_\theta$ is equal to $10^4$. To that end, the velocity profiles are integrated at small intervals along the plate and the profile where $Re_\theta$ is the closest to $10^4$ is selected. This target value is typically matched very closely. As shown in Fig. 11, all the velocity profiles are consistent with the wall function from their boundary node up

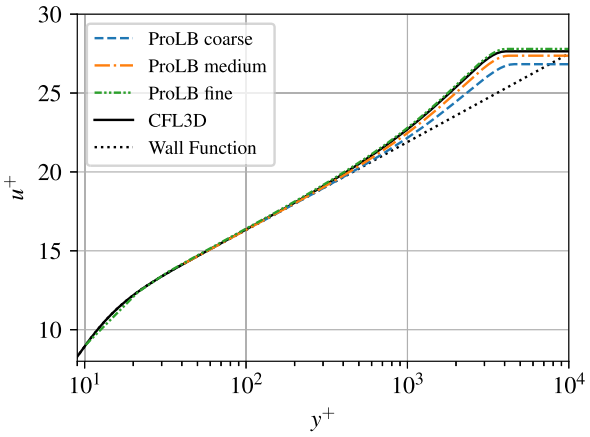


**Fig. 11.** Velocity profile at the location of $Re_\theta = 10^4$ for different grid spacings, compared to CFL3D results from NASA's Turbulence Modeling Resource [66].

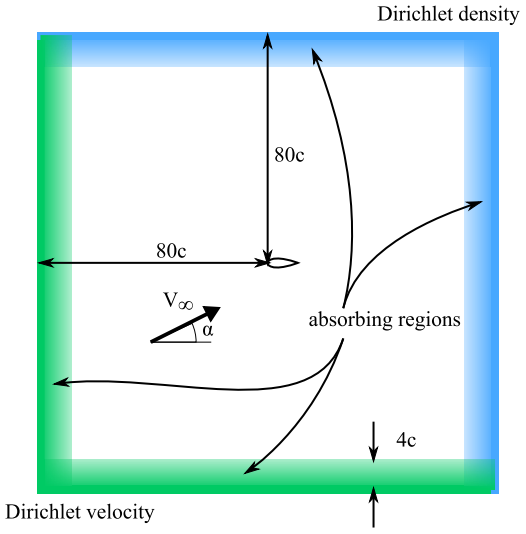


**Fig. 12.** Schematic of the computational set-up for the NACA0012 test case.

to about $y^+ = 200$, indicating that the wall function boundary condition is implemented consistently with the numerical scheme. The agreement between the profiles obtained in ProLB using the wall treatment and the reference computation with CFL3D is improved as the grid is refined, and it is very close for the fine grid.

Due to the isotropy of the grid and to the choice of a streamwise uniform construction, the grids used for this simple test case are much larger than those reported in [66] ($\approx 0.21 \times 10^6$ elements for the converged mesh): $\approx 1.5 \times 10^6$, $\approx 5.0 \times 10^6$, and $\approx 2.7 \times 10^7$ for the coarse, medium, and fine grids respectively. For a more efficient discretization, the grid at the wall could be selectively refined towards the upstream end of the plate, where the boundary layer is the thinnest. For the sake of simplicity and robustness, this option is not pursued in the present study.

### *4.2. NACA0012 airfoil*

#### *4.2.1. Test case presentation*

The symmetrical NACA0012 airfoil is a common test case in CFD for attached flow around a streamlined geometry [66,69] and LBM computations following different approaches have been reported. A VLES (Very Large Eddy Simulation) turbulence modeling approach has been applied —mostly in conjunction with a wall function— from $\mathrm{Re}_c = 500$ to $\mathrm{Re}_c = 5 \times 10^8$ [69–71]. RANS computations were performed using the S–A turbulence model at $\mathrm{Re}_c = 5 \times 10^5$ [72,73] with wall-resolved grids. It was also the support of DNS computations of laminar-turbulent transition [39] and for laminar flow computations [74].

The flow conditions documented in the NASA Turbulence Modeling Resource [66] are considered in the present study,: fully turbulent flow, $\mathrm{Ma} = 0.15$ and $\mathrm{Re}_c = 6.0 \times 10^6$. The original NACA0012 airfoil profile is extended to obtain a sharp trailing edge and then rescaled to a unit chord $c = 1$, thus the thickness of the airfoil is slightly reduced below its nominal 12% [66]. In comparison with the flat plate, the grid irregularity at the wall and the streamwise pressure gradients constitute additional challenges to the boundary layer flow modeling. A threshold wall distance value of $t = 1$ is used throughout this test case, meaning that all boundary nodes which wall distance is lower than the local grid spacing are eliminated.

A square computational domain is used, in which the airfoil leading edge is positioned at the centre, as shown in Fig. 12. Since the airfoil is symmetric, its chord is kept aligned with the grid axes to enable the generation of a symmetrical grid, and only positive angles of attack are considered. The angle of attack $\alpha$ is varied by modifying the free stream inflow conditions (Dirichlet condition on the velocity $V_\infty$) imposed on the upstream and lower faces of the domain, while the freestream density ($\rho_\infty$) is imposed on the downstream and upper boundaries for all computations. As these boundaries are reflective, absorbing regions are placed adjacent to them to damp acoustic waves inside the domain, which mainly originate from the initialization of the computation. Like for the flat plate case, the grid spacing is kept constant around the skin of the airfoil but it is coarsened away from it in layers of approximately 9 nodes each while ensuring the grid transition interfaces are symmetric. The domain is periodic in the spanwise direction and its width is equal to the far field grid spacing, as for the flat plate case. For the selected grid, its spanwise extent is 64 times the grid spacing on the airfoil.

#### *4.2.2. Grid effects*

To ensure that the far-field boundaries are sufficiently far from the airfoil for a negligible impact on the results in comparison with other sources of error, their distances are varied from $20c$ to $100c$ at the angle of attack of 10° and with a near wall grid spacing of $c/h = 1000$ nodes per chord length. The thickness of the absorbing layers is also varied in conjunction with the far-field distance, from $2c$ to $4c$. For simplicity, only the effects on the drag coefficient and lift coefficient are considered, where $\mathcal{D}$ and $\mathcal{L}$ are the integrated force components along and perpendicular to the freestream velocity respectively, and $S$ is the planform area of the extruded airfoil. The distance of the far-field boundaries seems to have a relatively low influence on $C_d$, with a variation of approximately 8% over the range considered, as shown in Fig. 13. As the drag only varies by $\Delta C_d = 0.0001$ between $80c$ and $100c$, the far-field distance of $80c$ is chosen for the remainder of the study, with $4c$ thick absorbing layers. No meaningful effect is observed on $C_l$.

The uniform grid spacing at the wall is also varied, from $c/h = 250$ to $c/h = 4000$ points per chord, at the angle of attack of 10°. The thickness of the refinement layers around the airfoil varies with grid spacing, since it is defined relative to the local grid spacing i.e. the growth rate is kept constant. With a drag variation of about 280% over the range considered, the grid density has a much larger influence than that of the far-field distance, as shown in Fig. 14. Since there remains a significant difference of $\Delta C_d = 0.0015$ between the finest ($c/h = 4000$) and second finest ($c/h = 3200$) grid, the finest one is selected for the remainder of the study. This grid density is finer than those reported in the previous studies using wall functions (approximately 500 points per

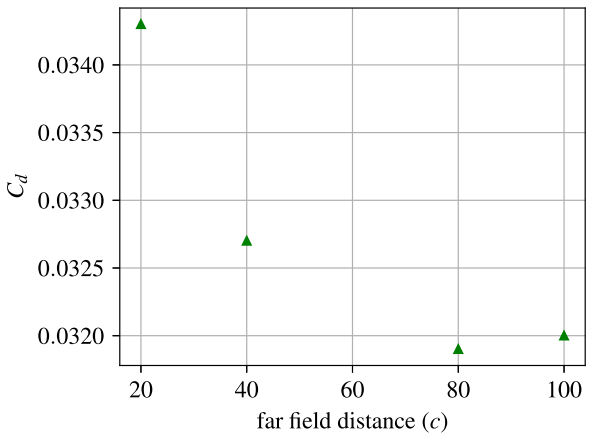


**Fig. 13.** Drag coefficient of the NACA0012 airfoil for different distances between the airfoil and the far-field boundaries of the computational domain at $\alpha = 10^\circ$ and $c/h = 1000$ nodes per chord.

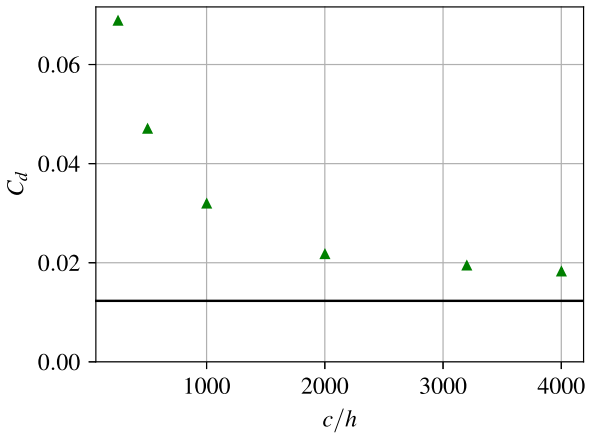


**Fig. 14.** Drag coefficient of the NACA0012 airfoil for different grid spacings at the skin, in nodes per chord at $\alpha = 10^\circ$. The horizontal line shows the level of the CLF3D result from NASA's Turbulence Modeling Resource [66].

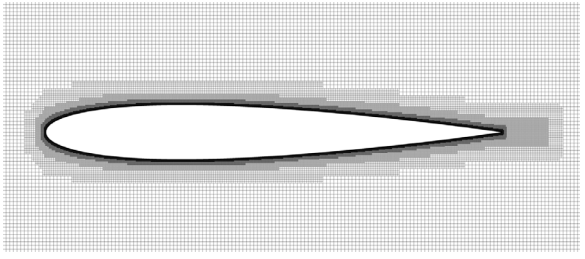

**Fig. 15.** Cross section of the selected grid ($c/h = 4000$ nodes per chord) around the NACA0012 airfoil, showing the 6 finest refinements.

chord [70] and 850 points per chord [71]), and it is coarser than those reported for the wall resolved ones for a lower Reynolds number (5000 points per chord [64] and 8192 points per chord [73]). A cross section of this grid in the vicinity of the airfoil is shown in Fig. 15. Although a strict grid convergence is not achieved, the error with respect to the reference computation is significantly reduced as the grid is refined. No significant effect is observed on the lift coefficient.

The $y^+$ of the boundary nodes on the suction side of the airfoil is mostly around 100, as shown in Fig. 16, which is appropriate for using a wall function as they are likely to be inside the inertial layer. It exceeds $y^+ = 400$ near the leading edge, indicating

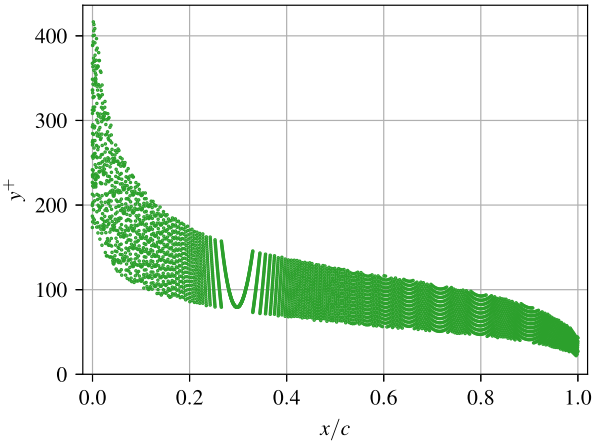


**Fig. 16.** Wall distance of boundary nodes in wall units $y^+$ on the suction side of the NACA0012 airfoil at $\alpha = 10^\circ$ for the selected grid (4000 nodes per chord).

that the grid is probably still too coarse for the donor nodes to be within the range of validity of the wall function in this area. Towards the trailing edge, it falls below $y^+ = 50$, such that some boundary nodes could be located in the buffer layer, suggesting that the grid is overly fine in this region. A more optimal distribution of nodes could be achieved if the grid density was decreased along the chord. The wide spread of the $y^+$ point cloud is due to the staircase pattern of the boundary nodes inherent to the use of a Cartesian grid as noted in Section 3.3. Around the location of maximum thickness ($x/c \approx 0.3$), the airfoil skin becomes tangent to the grid axis and in the absence of steps in the grid boundary, the $y^+$ of boundary nodes follows a parabola-like curve.

The selected grid comprises a total of $\approx 1.07 \times 10^8$ nodes, which is much larger than that for the reference computations with the CFL3D solver [66]. This is a consequence of the isotropy of the grid for achieving adequate $y^+$ at boundary nodes: near the wall, the streamwise and spanwise spacings are the same as the wall-normal one, even though there is no benefit for this with a RANS model. Additionally, since a 3D solver is used for this 2D case, the width of the domain is linked to the coarsest grid spacing. Indeed, increasing the far field grid spacing increases the width of the periodic domain, and thus the number of fine nodes near the wall in the spanwise direction. Due to this trade off, which is only valid for a thin slice computation, nodes far away from the airfoil account for a non-negligible fraction of the total computational cost. Finally the elimination of boundary nodes too close to walls further increases the $y^+$ for a given grid spacing $h$.

#### *4.2.3. Results*

The results obtained in ProLB with the present wall treatment are compared to the ones from the classic finite volume solvers CFL3D [75] and elsA [76]. The CFL3D results are retrieved from the NASA Turbulence Modeling Resource [66]. Previous studies on the same geometry with the Spalart–Allmaras turbulence model employing a wall function [70] or resolved wall boundary conditions [72,73] are not included in comparisons, as their computations were made at different Reynolds numbers.

Firstly, the surface coefficients on the airfoil skin are examined. The pressure coefficient is defined as

$$C_p = \frac{p - p_\infty}{1/2 \rho_\infty V_\infty^2} \tag{21}$$

with $p$ the static pressure, $p_\infty$ the far field static pressure, is shown in Figs. 17, 18, and 19 at $\alpha = 0^\circ, 10^\circ$ and $15^\circ$ respectively. It varies smoothly along the chord of the airfoil at all angles of attack considered, except for some minor oscillations on the suction peak. It

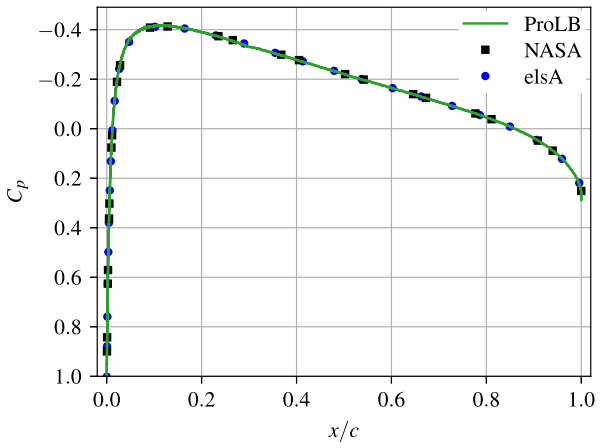


**Fig. 17.** Pressure coefficient $C_p$ along the chord of the NACA0012 airfoil at $\alpha = 0^\circ$, for ProLB with the present wall treatment, CFL3D [66], and elsA.

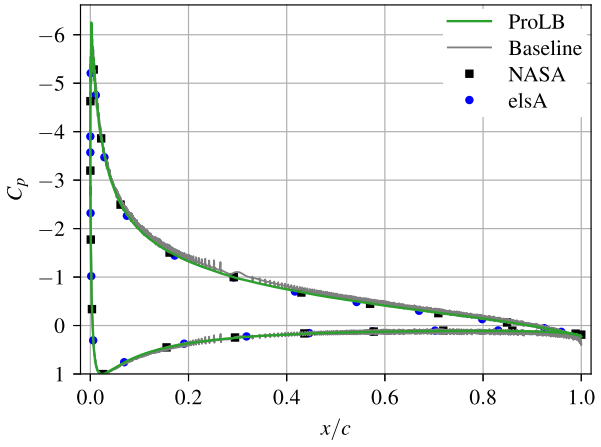


**Fig. 18.** Pressure coefficient $C_p$ along the chord of the NACA0012 airfoil at $\alpha = 10^\circ$, for ProLB with the present wall treatment, ProLB with the baseline wall treatment, CFL3D [66], and elsA.

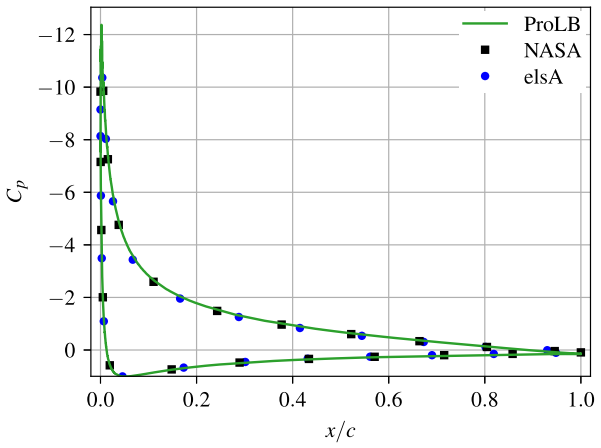


**Fig. 19.** Pressure coefficient $C_p$ along the chord of the NACA0012 airfoil at $\alpha = 15^\circ$, for ProLB with the present wall treatment, CFL3D [66], and elsA.

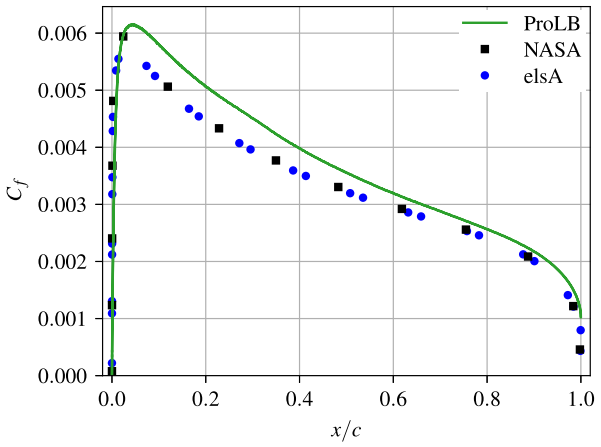


**Fig. 20.** Skin friction coefficient $C_f$ along the chord of the NACA0012 airfoil at $\alpha = 0^\circ$ for ProLB with the present wall treatment, CFL3D [66] (suction side only), and elsA.

is also in very close agreement with results obtained with classic RANS solvers on body-fitted meshes, except for a slight shift on the suction surface at 15°.

The skin friction coefficient $C_f$ (shown in Figs. 20, 21, and 22 for $\alpha = 0^\circ$, 10°, and 15° respectively) also varies smoothly along the chord, although the agreement with the classic RANS computations is not as good as for the surface pressure. At $\alpha = 0^\circ$, the surface coefficients on the upper and lower surfaces overlap. The agreement between the present results and the benchmark is progressively improved along the chord although the values of the body fitted computations are not reached. Like for the flat plate case, the error is large where the boundary layer is very thin and it is reduced as the boundary layer develops. The slight differences between the CFL3D and the elsA results are likely due to the differences between the meshes used, neither of which are elaborated to a calibration standard [66].

As the angle of attack is increased, the agreement on the pressure side is improved, suggesting a better boundary layer modeling. Indeed, as the stagnation point moves onto the pressure side, this side becomes more similar to a flat plate and the adverse pressure gradient observed at $\alpha = 0^\circ$ is eliminated. On the suction side, where the adverse pressure gradient becomes stronger with angle of attack, the agreement with the body-fitted computations improves towards the trailing edge, yet as at $\alpha = 0^\circ$, $C_f$ remains over-estimated. In the CFL3D and elsA computations at $\alpha = 15^\circ$, 2D flow separation occurs on the suction side close to $x/c = 0.9$, as $C_f$ reaches 0. This change in flow topology is not captured with the present wall treatment.

Besides the present improved wall treatment, Figs. 18 and 21 show the results with the baseline treatment for $\alpha = 10^\circ$. This baseline corresponds to the same solver (and identical simulation set-up), but with its 3 main features disabled: the advanced interpolation for $u_\tau$, the discrete wall function gradient evaluation, and the elimination of boundary nodes too close to the wall. Sharp wiggles are observed on the pressure profile with the baseline treatment, especially around $x/c \approx 0.3$, even though the overall values remain close to the references. The skin friction profiles obtained with the baseline treatment are noisier than the corresponding pressures, and they show much larger deviations with respect to the references. It thus appears that the proposed treatment is a great improvement over the baseline, both in terms of accuracy and of smoothness.

To further examine how the boundary layer is captured using the improved wall treatment, normal velocity profiles are extracted from the ProLB and elsA results for $\alpha = 10^\circ$ at different locations along the chord: close to the leading edge ($x/c = 0.1$), at midchord ($x/c = 0.5$), and close to the trailing edge ($x/c = 0.9$) on both the pressure side and the suction side.

On the pressure side, at $x/c = 0.1$ (Fig. 23), the elsA profile follows the wall function profile in the viscous sublayer and the

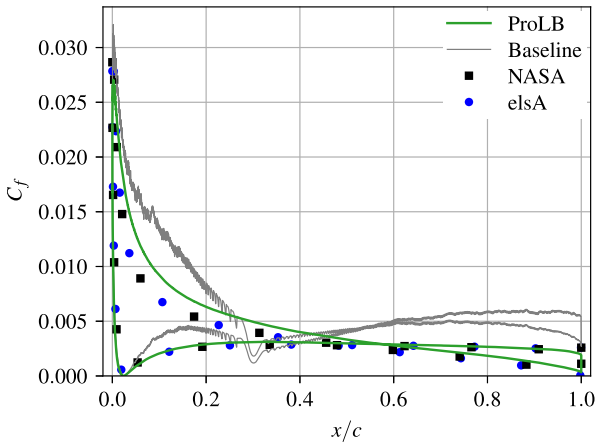


**Fig. 21.** Skin friction coefficient $C_f$ along the chord of the NACA0012 airfoil at $\alpha = 10^\circ$ for ProLB with the present wall treatment, ProLB with the baseline wall treatment, CFL3D [66], and elsA.

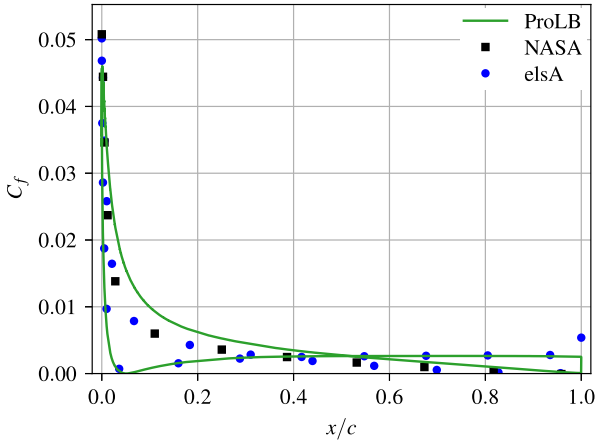


**Fig. 22.** Skin friction coefficient $C_f$ along the chord of the NACA0012 airfoil at $\alpha = 15^\circ$ for ProLB with the present wall treatment, CFL3D [66] (suction side only), and elsA.

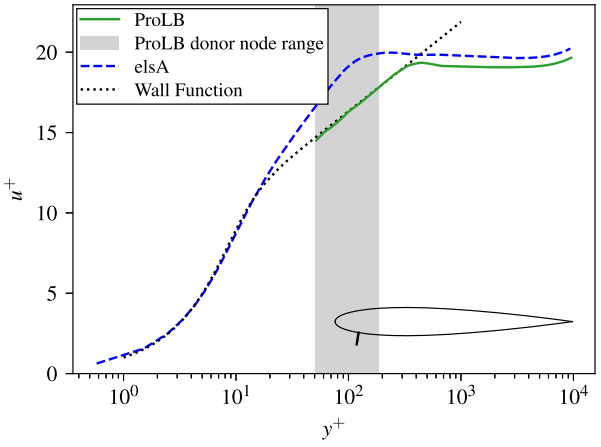


**Fig. 23.** Normal velocity profile in wall units extracted at $x/c = 0.1$ on the pressure side of the airfoil at $\alpha = 10^\circ$, for ProLB with the present wall treatment and elsA, compared to the Spalart–Allmaras wall function. The grey box shows the potential wall distance range of the donor nodes in ProLB.

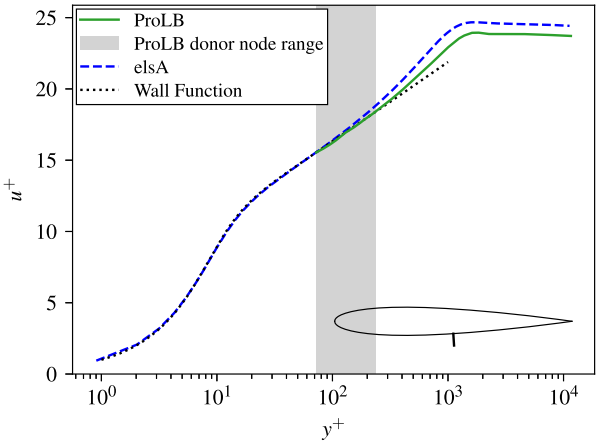


**Fig. 24.** Normal velocity profile in wall units extracted at $x/c = 0.5$ on the pressure side of the airfoil at $\alpha = 10^\circ$, for ProLB with the present wall treatment and elsA, compared to the Spalart–Allmaras wall function. The grey box shows the potential wall distance range of the donor nodes in ProLB.

buffer layer up to about $y^+ = 15$. Above, however, it appears there is no logarithmic region, as observed by Knopp et al. [55] close to the leading edge of another airfoil. A low friction Reynolds number ($\mathrm{Re}_\tau = \frac{u_\tau \delta}{\nu}$ with $\delta$ the boundary layer thickness) of $\mathrm{Re}_\tau = 151$ in elsA confirms that the RANS boundary layer is not very developed at this location, only about $\Delta x/c \approx 0.03$ downstream from the stagnation point. For the boundary layer to be accurately captured using the wall treatment, the boundary nodes and their donor nodes need to be located within the range of validity of the wall model i.e. the range in which there is a good agreement between wall model and the RANS solution. At this location, the grid spacing would have to be about 13 times smaller than the current size, with the boundary nodes in the viscous sublayer and the donor nodes in the buffer layer. Although it is not accurate, the velocity profile obtained with the wall treatment is consistent with the wall function profile in the entire boundary layer, thus bypassing the outer region of the boundary layer. Although this is not physically coherent, it appears to be the consequence of imposing the wall function profile where the boundary layer is too thin to be resolved.

Further downstream on the pressure side, at $x/c = 0.5$ (Fig. 24), the boundary layer is further developed in elsA, with $\mathrm{Re}_\tau$ increased to 1663, and the wall function validity is extended up to approximately $y^+ = 80$. The profile obtained with the present wall treatment is much closer to elsA, but even though the velocity at the boundary node is the same, differences exist in the wake region. Indeed, as the boundary node is located at the upper bound of the wall function validity, the donor nodes are located above it, inevitably introducing errors.

At $x/c = 0.9$ (Fig. 25), the boundary layer continues thickening ($Re_\tau = 2892$ in elsA) and the wall function remains valid until about $y^+ = 200$. The wall treatment yields a profile in good agreement with the one obtained with elsA. It appears that the errors in the boundary layer history are drowned out by the thickening of the boundary layer as it develops, such that an accurate solution can be found even though it is under resolved farther upstream.

On the suction surface, at $x/c = 0.1$ (Fig. 26), the boundary layer is already more developed than at the same location on the pressure surface ($\mathrm{Re} = 1338$ in elsA), with the wall function valid up to about $y^+ = 100$. Similarly as on the pressure surface, however, the profile obtained with the wall treatment is consistent with the wall function profile, yet far from the elsA result. Indeed the boundary node and the donor nodes are above the validity range of the wall law.

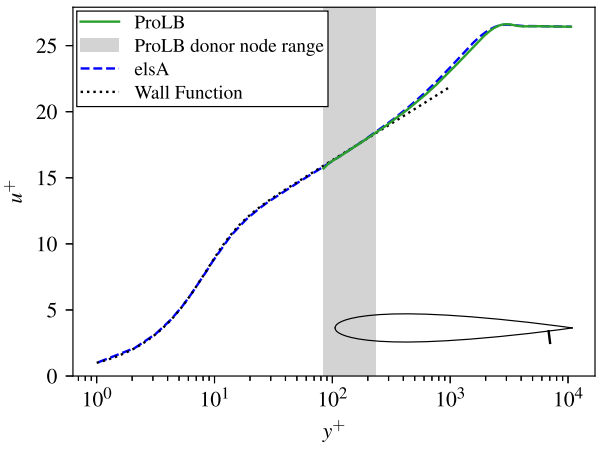


**Fig. 25.** Normal velocity profile in wall units extracted at $x/c = 0.9$ on the pressure side of the airfoil at $\alpha = 10^\circ$, for ProLB with the present wall treatment and elsA, compared and to the Spalart–Allmaras wall function. The grey box shows the potential wall distance range of the donor nodes in ProLB.

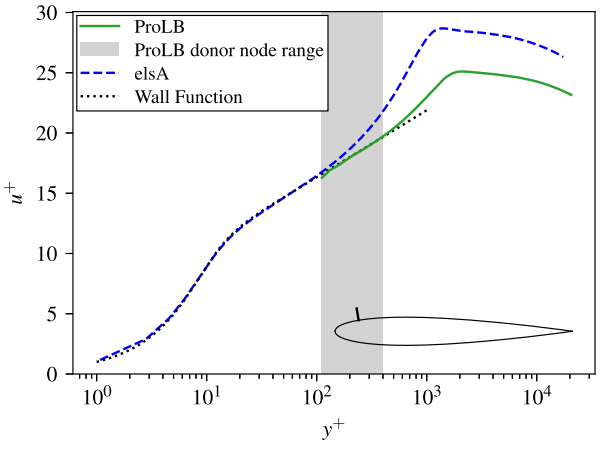


**Fig. 26.** Normal velocity profile in wall units extracted at $x/c = 0.1$ on the suction side of the airfoil at $\alpha = 10^\circ$, for ProLB with the present wall treatment and elsA, compared to the Spalart–Allmaras wall function. The grey box shows the potential wall distance range of the donor nodes in ProLB.

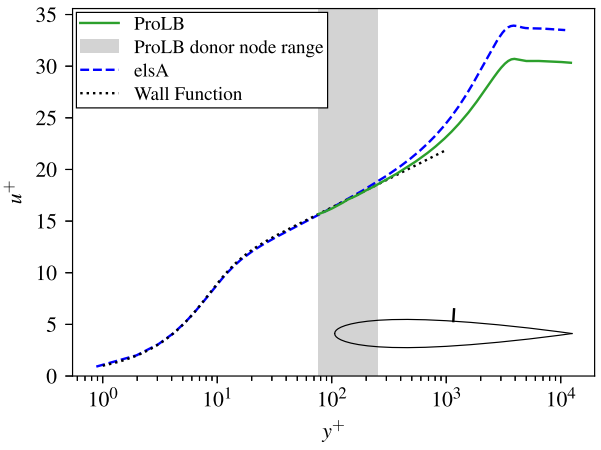


**Fig. 27.** Normal velocity profile in wall units extracted at $x/c = 0.5$ on the suction side of the airfoil at $\alpha = 10^\circ$, for ProLB with the present wall treatment and elsA, compared to the Spalart–Allmaras wall function. The grey box shows the potential wall distance range of the donor nodes in ProLB.

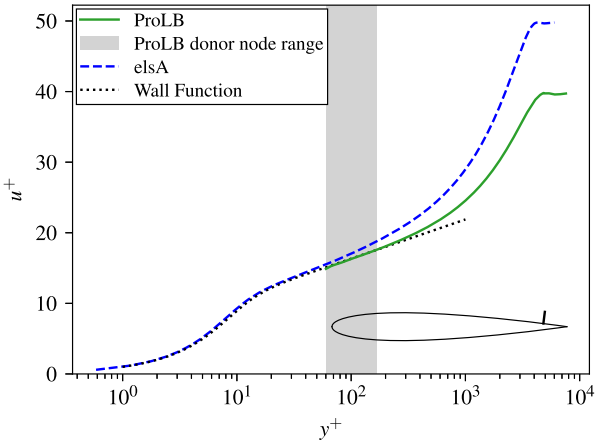


**Fig. 28.** Normal velocity profile in wall units extracted at $x/c = 0.9$ on the suction side of the airfoil at $\alpha = 10^\circ$, for ProLB with the present wall treatment and elsA, compared to the Spalart–Allmaras wall function. The grey box shows the potential wall distance range of the donor nodes in ProLB.

At $x/c = 0.5$ (Fig. 27), the wall function seems valid up to nearly $y^+ = 200$ yet the slope of the logarithmic region seems higher in elsA than in the wall function. This is likely because of the adverse pressure gradient, consistently with the profiles shown by Knopp et al. [55]. With the wall treatment, even though the donor nodes are within the range of validity of the wall function, and despite the higher friction Reynolds number ($\mathrm{Re}_\tau \approx 3802$ in elsA) than at $x/c = 0.9$ on the pressure side, the agreement between the ProLB and the elsA profiles is not significantly better than at $x/c = 0.1$.

At $x/c = 0.9$ (Fig. 28), the wall function no longer seems to be applicable. Indeed the slope of the inertial layer seems further increased than at $x/c = 0.5$ and a small shift is visible, even in the buffer layer, as the boundary layer is deformed by the adverse pressure gradient and approaches separation. The agreement between the present wall treatment and elsA is worse than at $x/c = 0.5$ despite the larger friction Reynolds number ($\mathrm{Re}_\tau \approx 4223$ in elsA).

Finally the global lift and drag coefficients are considered, which are both integrated form the surface coefficients. The lift coefficient predicted with ProLB using the present wall treatment is in very good agreement with the body-fitted RANS codes and thin airfoil theory in the linear part of the lift curve ($\alpha \lessapprox 10^\circ$), as shown in Fig. 29, in line with the very good agreement of the pressure coefficients. Viscous effects are expected to have minimal contributions to the lift in this range, so that the errors in boundary layer modeling on the suction surface have little impact on the lift.

At higher angles of attack ($\alpha \gtrapprox 10^\circ$), however, viscous effects such as trailing edge flow separation on the suction surface, reduce the lift in comparison to thin airfoil theory. This non-linearity is under-estimated as the accuracy of the present wall treatment is reduced. Indeed at $\alpha = 15^\circ$ the local flow separation on the suction side in CFL3D and in elsA is not captured with the present wall treatment in conjunction with the S–A wall function, as noted in Fig. 22. Since 2D RANS computations cannot be expected to be reliable in predicting stall, wind tunnel results are added for further reference. At $\alpha = 20^\circ$, stall has already occurred both in the wind tunnel and in the elsA computation but the lift predicted with ProLB continues to increase as massive flow separation is delayed.

At $\alpha = 0^\circ$, the drag coefficient ($C_d$) estimated with the present wall treatment is in good agreement with the other data, as shown in Fig. 30. Similarly to the lift, the results degrade as the angle of attack is increased and the drag is over-estimated, as it is the

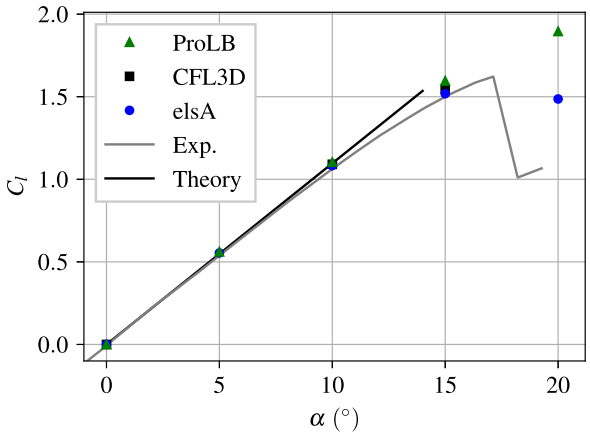


**Fig. 29.** Lift coefficient ($C_l$) as a function of angle of attack for ProLB with the present wall treatment, CFL3D [66], elsA, and experimental results [77].

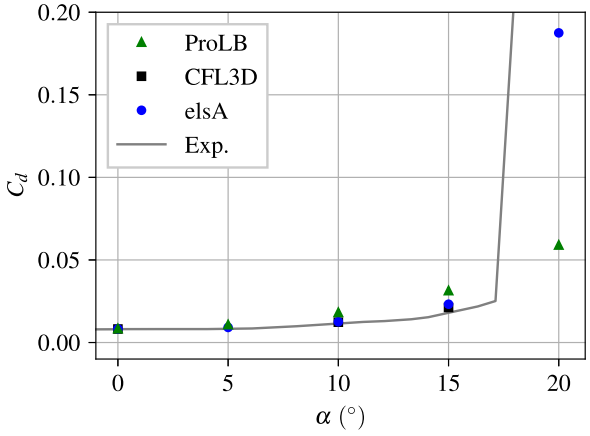


**Fig. 30.** Drag coefficient ($C_d$) as a function of angle of attack for ProLB with the present wall treatment, CFL3D [66], elsA, and experimental results [77].

case for the $C_f$ on the suction side. On the contrary, at $\alpha = 20°$, the sudden drag rise due to stall is not captured.

The inability of the present implementation to predict flow separation (and thus stall) accurately is its major limitation. It is not surprising, however, since the simple algebraic wall function used does not take the effects of adverse pressure gradients into account.

## 5. Conclusion

An improved immersed boundary turbulent wall treatment for high Reynolds number flows on Cartesian grids has been presented in the context of LBM. The under-resolved velocity gradients near the wall are carefully taken into account by imposing a discretized form of the gradients from the wall model. Boundary nodes located too close to the wall with respect to the grid spacing to be accurately computed by the numerical scheme are eliminated, thus locally shifting the application of the boundary condition away from the wall where appropriate.

This treatment has been validated through URANS computations (focusing on steady solutions) with a node-based LBM solver on a flat plate with a body-fitted grid and on a NACA0012 airfoil. In both cases, visibly smooth surface pressure and skin friction coefficients have been obtained. Furthermore, its potential for accurately capturing the boundary layer on a non-body-fitted curved surface has been demonstrated, provided the underlying wall model remains valid. In comparison with the baseline wall treatment, significant improvements in smoothness and accuracy of surface coefficients is demonstrated. With the algebraic wall function used in this study, surface coefficients were accurately predicted where the boundary layer was well developed, its outer region was sufficiently resolved, and in the absence of significant adverse pressure gradients. Flow separation and stall are not well predicted.

Future studies should investigate the potential improvements in physical modeling that could be achieved through the integration of a more advanced wall function including streamwise pressure gradient effects. Since this wall treatment has been validated for attached flow with a RANS model, it could be employed in computations with hybrid RANS-LES models on more complex geometries.

## Declaration of Competing Interest

None.

## CRediT authorship contribution statement

**Johan Degrigny:** Conceptualization, Methodology, Software, Validation, Formal analysis, Writing - original draft. **Shang-Gui Cai:** Conceptualization, Methodology, Software, Writing - review & editing. **Jean-François Boussuge:** Project administration, Supervision, Resources, Software, Writing - review & editing. **Pierre Sagaut:** Project administration, Supervision, Resources, Software, Writing - review & editing.

## Acknowledgements

This work was supported by Airbus Operations and ANRT through the CIFRE scheme. J.D. is grateful to G. Wissocq and C. Gacherieu for fruitful discussions that have insipred some of the developments. Post-processing for this work was performed using the antares python library (https://www.cerfacs.fr/antares).

## References

[1] Piomelli U, Balaras E. Wall-layer models for large-eddy simulations. Annu Rev Fluid Mech 2002;34(1):349–74. doi:10.1146/annurev.fluid.34.082901.144919.
[2] Bose ST, Park GI. Wall-modeled large-eddy simulation for complex turbulent flows. Annu Rev Fluid Mech 2018;50(1):535–61. doi:10.1146/annurev-fluid-122316-045241.
[3] Chapman DR. Computational aerodynamics development and outlook. AIAA J 1979;17(12):1293–313. doi:10.2514/3.61311.
[4] Choi H, Moin P. Grid-point requirements for large eddy simulation: Chapman's estimates revisited. Phys Fluids 2012;24(1):011702. doi:10.1063/1.3676783.
[5] Launder BE, Spalding DB. The numerical computation of turbulent flows. Comput Methods Appl Mech Eng 1974;3(2):269–89. doi:10.1016/0045-7825(74)90029-2.
[6] Peskin CS. Flow patterns around heart valves: a numerical method. J Comput Phys 1972;10(2):252–71. doi:10.1016/0021-9991(72)90065-4.
[7] Mittal R, Iaccarino G. Immersed boundary methods. Annu Rev Fluid Mech 2005;37(1):239–61. doi:10.1146/annurev.fluid.37.061903.175743.
[8] Mohd-Yusof J. Combined immersed-boundary/b-spline methods for simulations of flow in complex geometries. Center Turbul Res Annual Res Briefs 1997 1997.
[9] Fadlun E, Verzicco R, Orlandi P, Mohd-Yusof J. Combined immersed-boundary finite-difference methods for three-dimensional complex flow simulations. J Comput Phys 2000;161(1):35–60. doi:10.1006/jcph.2000.6484.
[10] Tessicini F, Iaccarino G, Fatica M, Wang M, Verzicco R. Wall modeling for large-eddy simulation using an immersed boundary method. Center Turbul Res Ann Res Briefs 2002 2002:7.
[11] Kalitzin G, Iaccarino G. Turbulence modeling in an immersed-boundary RANS method. Center Turbul Res Annu Res Briefs 2002 2002:12.
[12] Kalitzin G, Iaccarino G. Toward immersed boundary simulation of high reynolds number flows. Center Turbul Res Ann Res Briefs 2003 2003:10.
[13] Ma M, Huang W-X, Xu C-X. A dynamic wall model for large eddy simulation of turbulent flow over complex/moving boundaries based on the immersed boundary method. Phys Fluids 2019;31(11):115101. doi:10.1063/1.5126853.
[14] Roman F, Armenio V, Fröhlich J. A simple wall-layer model for large eddy simulation with immersed boundary method. Phys Fluid 2009;21(10):101701. doi:10.1063/1.3245294.

[15] Kang S. An improved near-wall modeling for large-eddy simulation using immersed boundary methods. Int J Numer Methods Fluids 2015;78(2):76–88. doi:10.1002/fld.4008.

[16] Malaspinas O, Sagaut P. Wall model for large-eddy simulation based on the lattice Boltzmann method. J Comput Phys 2014;275:25–40. doi:10.1016/j.jcp.2014.06.020.

[17] Wilhelm S, Jacob J, Sagaut P. An explicit power-law-based wall model for lattice boltzmann method–Reynolds-averaged numerical simulations of the flow around airfoils. Phys Fluid 2018;30(6):065111. doi:10.1063/1.5031764.

[18] Haussmann M, Claro Barreto A, Lipeme Kouyi G, Rivière N, Nirschl H, Krause MJ. Large-eddy simulation coupled with wall models for turbulent channel flows at high Reynolds numbers with a lattice Boltzmann method — application to Coriolis mass flowmeter. Comput Math Appl 2019;78(10):3285–302. doi:10.1016/j.camwa.2019.04.033.

[19] Maeyama H, Imamura T, Osaka J, Kurimoto N. Turbulent channel flow simulations using the lattice Boltzmann method with near-wall modeling on a non-body-fitted Cartesian grid. Computers & Mathematics with Applications 2021;93:20–31. doi:10.1016/j.camwa.2021.04.003.

[20] Lee J-d, Ruffin S. Development of a turbulent wall-function based viscous Cartesian-grid methodology. 45th AIAA Aerospace Sciences Meeting and Exhibit. Reno, Nevada: American Institute of Aeronautics and Astronautics; 2007. doi:102514/62007-1326.

[21] Tamaki Y, Harada M, Imamura T. Near-wall modification of Spalart–Allmaras turbulence model for immersed boundary method. AIAA J 2017;55(9):3027–39. doi:10.2514/1.J055824.

[22] Nishimura S, Hayashi K, Nakaye S, Yoshimoto M, Suga K, Inamuro T. Implicit large-eddy simulation of rotating and non-rotating machinery with cumulant lattice Boltzmann method aiming for industrial applications. AIAA Aviation 2019 Forum. Dallas, Texas: American Institute of Aeronautics and Astronautics; 2019.

[23] Capizzano F. Turbulent wall model for immersed boundary methods. AIAA J 2011;49(11):2367–81. doi:10.2514/1.J050466.

[24] Zhou C. RANS simulation of high-Re turbulent flows using an immersed boundary method in conjunction with wall modeling. Comput Fluid 2017;143:73–89. doi:10.1016/j.compfluid.2016.11.009.

[25] Yu D, Mei R, Shyy W. A unified boundary treatment in lattice Boltzmann method. 41st Aerospace sciences meeting and exhibit. American institute of aeronautics and astronautics; 2003.

[26] Pasquali A, Geier M, Krafczyk M. Near-wall treatment for the simulation of turbulent flow by the cumulant lattice Boltzmann method. Comput Math Appl 2020;79(1):195–212. doi:10.1016/j.camwa.2017.11.022.

[27] Hoffman J. Simulation of turbulent flow past bluff bodies on coarse meshes using general Galerkin methods: drag crisis and turbulent euler solutions. Comput Mech 2006;38(4–5):390–402. doi:10.1007/s00466-006-0053-x.

[28] Geier M, Schönherr M, Pasquali A, Krafczyk M. The cumulant lattice Boltzmann equation in three dimensions: theory and validation. Comput Math Appl 2015;70(4):507–47. doi:10.1016/j.camwa.2015.05.001.

[29] Filippova O, Succi S, Mazzocco F, Arrighetti C, Bella G, Hänel D. Multiscale lattice Boltzmann schemes with turbulence modeling. J Comput Phys 2001;170(2):812–29. doi:10.1006/jcph.2001.6764.

[30] Filippova O, Hänel D. Grid refinement for lattice-BGK models. J Comput Phys 1998;147(1):219–28. doi:10.1006/jcph.1998.6089.

[31] Chen H, Teixeira C, Molvig K. Realization of fluid boundary conditions via discrete Boltzmann dynamics. Int J Modern Phys C 1998;09(08):1281–92. doi:10.1142/S0129183198001151.

[32] Teixeira CM. Incorporating turbulence models into the lattice-Boltzmann method. Int J Modern Phys C 1998;09(08):1159–75. doi:10.1142/S0129183198001060.

[33] Schneider A. A consistent large eddy approach for lattice Boltzmann methods and its application to complex flows. Technische Universität Kaiserslautern; 2015.

[34] König B, Fares E. Exa PowerFLOW simulations for the sixth AIAA drag prediction workshop. J Aircr 2018;55(4):1482–90. doi:10.2514/1.C034480.

[35] Konig B, Fares E, Murayama M, Ito Y. PowerFLOW Simulations for the Third AIAA High-Lift Prediction Workshop. 2018 AIAA aerospace sciences meeting. Kissimmee, Florida: American Institute of Aeronautics and Astronautics; 2018. ISBN 978-1-62410-524-1. doi:102514/62018-1255.

[36] Khorrami MR, Fares E. Simulation-based airframe noise prediction of a full-scale, full aircraft. In: 22nd AIAA/CEAS aeroacoustics conference. Lyon, France: American Institute of Aeronautics and Astronautics; 2016. ISBN 978-1-62410-386-5. https://doi.org/10.2514/6.2016-2706.

[37] Casalino D, Hazir A, Mann A. Turbofan broadband noise prediction using the lattice Boltzmann method. AIAA J 2018;56(2):609–28. doi:10.2514/1.J055674.

[38] van der Velden WCP, Casalino D, Gopalakrishnan P, Jammalamadaka A, Li Y, Zhang R, et al. Validation of jet noise simulations and resulting insights of acoustic near field. AIAA J 2019;57(12):5156–67. doi:10.2514/1.J057970.

[39] Ribeiro AFP, Casalino D, Fares E, Choudhari M. Direct numerical simulation of an airfoil with sand grain roughness on the leading edge. Technical Memorandum. Hampton, VA 23681-2199: NASA Langley Research Cente; 2016.

[40] Haussmann M, Ries F, Jeppener-Haltenhoff JB, Li Y, Schmidt M, Welch C, et al. Evaluation of a near-wall-modeled large eddy lattice Boltzmann method for the analysis of complex flows relevant to IC engines. Computation 2020;8(2):43. doi:10.3390/computation8020043.

[41] Bouzidi M, Firdaouss M, Lallemand P. Momentum transfer of a Boltzmann-lattice fluid with boundaries. Phys Fluid 2001;13(11):3452–9. doi:10.1063/1.1399290.

[42] Spalart PR. Detached-eddy simulation. Annu Rev Fluid Mech 2008;41(1):181–202. doi:10.1146/annurev.fluid.010908.165130.

[43] Bhatnagar PL, Gross EP, Krook M. A model for collision processes in gases. i. small amplitude processes in charged and neutral one-component systems. Phys Rev 1954;94(3):511–25. doi:10.1103/PhysRev.94.511.

[44] Jacob J, Malaspinas O, Sagaut P. A new hybrid recursive regularised Bhatnagar–Gross–Krook collision model for lattice Boltzmann method-based large eddy simulation. J Turbul 2018;19(11–12):1051–76. doi:10.1080/14685248.2018.1540879.

[45] Krüger T, Kusumaatmaja H, Kuzmin A, Shardt O, Silva G, Viggen EM. The lattice boltzmann method: principles and practice. graduate texts in physics. Cham: Springer International Publishing; 2017. doi:101007/978-3-319-44649-3. ISBN 978-3-319-44647-9 978-3-319-44649-3

[46] Spalart PR, Allmaras SR. One-equation turbulence model for aerodynamic flows. La Recherche Aerospatiale 1994(1):5–21. doi:10.2514/6.1992-439.

[47] Allmaras SR, Johnson FT, Spalart PR. Modifications and clarifications for the implementation of the Spalart-Allmaras turbulence model. Comput Fluid Dyn 2012:11.

[48] Chang P-H, Liao C-C, Hsu H-W, Liu S-H, Lin C-A. Simulations of laminar and turbulent flows over periodic hills with immersed boundary method. Comput Fluid 2014;92:233–43. doi:10.1016/j.compfluid.2013.10.043.

[49] Park H, Oh G, Park TS, Lee C, Choi J-I. An immersed boundary formulation incorporating a two-layer wall model approach for RANS simulations with complex geometry. Comput Fluid 2020;205:104551. doi:10.1016/j.compfluid.2020.104551.

[50] Reichardt H. Vollständige Darstellung der turbulenten Geschwindigkeitsverteilung in glatten Leitungen. ZAMM - Zeitschrift für Angewandte Mathematik und Mechanik 1951;31(7):208–19. doi:10.1002/zamm.19510310704.

[51] Spalding DB. A single formula for the law of the wall. J Appl Mech 1961;28(3):455–8.

[52] Musker AJ. Explicit expression for the smooth wall velocity distribution in a turbulent boundary layer. AIAA J 1979;17(6):655–7. doi:10.2514/3.61193.

[53] Bernardini M, Modesti D, Pirozzoli S. On the suitability of the immersed boundary method for the simulation of high-Reynolds-number separated turbulent flows. Comput Fluid 2016;130:84–93. doi:10.1016/j.compfluid.2016.02.018.

[54] Kalitzin G, Medic G, Iaccarino G, Durbin P. Near-wall behavior of RANS turbulence models and implications for wall functions. J Comput Phys 2005;204(1):265–91. doi:10.1016/j.jcp.2004.10.018.

[55] Knopp T, Alrutz T, Schwamborn D. A grid and flow adaptive wall-function method for RANS turbulence modelling. J Comput Phys 2006;220(1):19–40. doi:10.1016/j.jcp.2006.05.003.

[56] Latt J, Chopard B, Malaspinas O, Deville M, Michler A. Straight velocity boundaries in the lattice Boltzmann method. Phys Rev E 2008;77(5):056703. doi:10.1103/PhysRevE.77.056703.

[57] Verschaeve JC, Müller B. A curved no-slip boundary condition for the lattice Boltzmann method. J Comput Phys 2010;229(19):6781–803. doi:10.1016/j.jcp.2010.05.022.

[58] Astoul T, Wissocq G, Boussuge J-F, Sengissen A, Sagaut P. Analysis and reduction of spurious noise generated at grid refinement interfaces with the lattice Boltzmann method. J Comput Phys 2020;418. doi:10.1016/j.jcp.2020.109645.

[59] Harada M, Tamaki Y, Takahashi Y, Imamura T. Simple and robust cut-cell method for high-Reynolds-number-flow simulation on Cartesian grids. AIAA J 2017;55(8):2833–41. doi:10.2514/1.J055343.

[60] Cai S-G, Degrigny J, Boussuge J-F, Sagaut P. Coupling of turbulence wall models and immersed boundaries on Cartesian grids. J Comput Phys 2021;429:109995. doi:10.1016/j.jcp.2020.109995.

[61] Werner H, Wengle H. Large-eddy simulation of turbulent flow over and around a cube in a plate channel. In: Durst F, Friedrich R, Launder BE, Schmidt FW, Schumann U, Whitelaw JH, editors. Turbulent Shear Flows 8. Berlin, Heidelberg: Springer; 1993. p. 155–68. ISBN 978-3-642-77674-8. doi:10.1007/978-3-642-77674-8_12.

[62] Wilhelm S, Jacob J, Sagaut P. A new explicit algebraic wall model for LES of turbulent flows under adverse pressure gradient. Flow Turbul Combust 2020. doi:10.1007/s10494-020-00181-7.

[63] Guo Z, Zheng C, Shi B. An extrapolation method for boundary conditions in lattice Boltzmann method. Phys Fluid 2002;14(6):2007–10. doi:10.1063/1.1471914.

[64] Li Y, Zhang R, Shock R, Chen H. Prediction of vortex shedding from a circular cylinder using a volumetric lattice-Boltzmann boundary approach. Eur Phys J Spec Topic 2009;171(1):91–7. doi:10.1140/epjst/e2009-01015-9.

[65] Bocquet S, Ricot D, Sengissen A, Vincent-Viry C, Demory B, Henner M, Ailloud F. Evaluation of the lattice boltzmann method for aero-acoustic simulations of industrial air systems. In: 25th AIAA/CEAS Aeroacoustics Conference. Delft, The Netherlands: American Institute of Aeronautics and Astronautics; 2019. ISBN 978-1-62410-588-3. https://doi.org/10.2514/6.2019-2415.

[66] Rumsey C, Smith B, Huang G. Description of a website resource for turbulence modeling verification and validation. In: 40th Fluid Dynamics Conference and Exhibit. Chicago, Illinois: American Institute of Aeronautics and Astronautics; 2010. https://doi.org/10.2514/6.2010-4742.

[67] Xu H, Sagaut P. Analysis of the absorbing layers for the weakly-compressible lattice Boltzmann methods. J Comput Phys 2013;245:14–42. doi:10.1016/j.jcp.2013.02.051.

[68] Feng Y, Guo S, Jacob J, Sagaut P. Solid wall and open boundary conditions in hybrid recursive regularized lattice Boltzmann method for compressible flows. Phys Fluid 2019;31(12):126103. doi:10.1063/1.5129138.

[69] Roy CJ, Rumsey CL, Tinoco EN. Summary data from the sixth AIAA computational fluid dynamics drag prediction workshop: code verification. J Aircr 2018;55(4):1338–51. doi:10.2514/1.C034856.
[70] Lockard DP, Luo L-S, Milder SD, Singer BA. Evaluation of PowerFLOW for aerodynamic applications. J Stat Phys 2002;107(1):423–78. doi:10.1023/A:1014539411062.
[71] Shock RA, Mallick S, Chen H, Yakhot V, Zhang R. Recent results on two-dimensional airfoils using a lattice Boltzmann-based algorithm. J Aircr 2002;39(3):434–9. doi:10.2514/2.2969.
[72] Li K, Zhong C, Zhuo C, Cao J. Non-body-fitted Cartesian-mesh simulation of highly turbulent flows using multi-relaxation-time lattice Boltzmann method. Comput Math Appl 2012;63(10):1481–96. doi:10.1016/j.camwa.2012.03.080.
[73] Pellerin N, Leclaire S, Reggio M. An implementation of the Spalart–Allmaras turbulence model in a multi-domain lattice Boltzmann method for solving turbulent airfoil flows. Comput Math Appl 2015;70(12):3001–18. doi:10.1016/j.camwa.2015.10.006.
[74] Di Ilio G, Chiappini D, Ubertini S, Bella G, Succi S. Fluid flow around NACA 0012 airfoil at low-Reynolds numbers with hybrid lattice Boltzmann method. Comput Fluid 2018;166:200–8. doi:10.1016/j.compfluid.2018.02.014.
[75] Rumsey C.. CFL3D Home Page. https://cfl3d.larc.nasa.gov/; 2017.
[76] ONERA. ONERA elsA. http://elsa.onera.fr/; 2018.
[77] Ladson CL. Effects of independent variation of Mach and Reynolds numbers on the low-speed aerodynamic characteristics of the NACA 0012 airfoil section. Technical Memorandum. Hampton, VA, United States: NASA Langley Research Center; 1988.