POLITECNICO DI TORINO

Corso di Laurea Magistrale

in Ingegneria Aerospaziale

Tesi di Laurea Magistrale

Development and Validation of an improved wall-function boundary condition for computational aerodynamics

Relatori
prof. Domenic D'Ambrosio
prof. Stefan Wallin

Candidato
Carlo Loris Palombo

Anno Accademico 2020/2021

Development and validation of an improved wall-function boundary condition for computational aerodynamics

Abstract

Computational Fluid Dynamics is a powerful and widely used tool for developing projects that concern flow motion, in very different fields. Industrial CFD solvers are continuously developed with the aim of improving accuracy and reducing the computational cost of the simulations. Wall-flow cases are particular demanding as the presence of a solid-surface interface generates steep gradients in the proximity of the wall. Resolving such gradients can be crucial to obtain a consistent solution but also very expensive in terms of grid refinement, and hence computational time. Wall functions are widely used and offer significant computational savings when it comes to near-wall flow resolution. Previous wall function implemented in the M-Edge solver suffered by poor performances in complex flows characterized by strong pressure-gradient phenomena, such as separation. A new formulation has been developed and validated for $k - \omega$ and Spalart-Allmaras models. Test simulations started from simple and near-ideal cases (2D zero pressure gradient flat plate) and advanced to always more complex flow cases and geometries (full 3D general fighter). Every case has been run coupling the wall-function boundary condition with three different turbulence models: the Menter SST, the Menter BSL with a EARSM closure and the Spalart-Allmaras one-equation model. Overall results showed the upgraded performance of new wall function in flow resolution together with more agile grid requirements, faster and deeper convergence of the residuals and a general reduction in computational time.

Keywords

Wall Functions, Wall Treatment, M-Edge, Computational Fluid Dynamics, Turbulent Boundary Layer, Turbulence Models.

Sommario

La Fluidodinamica computazionale (CFD) è un potente strumento, ampiamente diffuso nello sviluppo di progetti che coinvolgano il movimento di fluidi, in svariati ambiti. I solutori CFD in ambito industriale sono continuamente migliorati con l'intento di aumentare l'accuratezza e ridurre i costi computazionali delle simulazioni. I casi di flussi a parete sono particolarmente sensibili dato che la presenza di una superficie solida di interfaccia genera gradienti elevati in prossimità della parete. Risolvere opportunamente questi gradienti può risultare vincolante al fine di ottenere una soluzione consistente ma al contempo molto costosa in termini di raffinamento della griglia, e quindi di tempo computazionale. Le funzioni di parete sono ampiamente usate e offrono risparmi computazionali significanti nel caso di flussi a parete. La precedente funzione di parete implementata nel solutore M-Edge presentava performance basse in flussi complessi caratterizzati da fenomeni di forti gradienti di pressione, come la separazione. Una nuova formulazione è stata sviluppata e validata per modelli turbolenti $k-\omega$ e Spalart-Allmaras. Le simulazione di test sono partite da casi semplici e quasi-ideali (lamina piana 2D con gradiente di pressione nullo) e sono avanzate con casi e geometrie sempre più complessi (fighter generico 3D). Ogni caso è stato simulato implementando la funzione di parete con tre diversi modelli di turbolenza: il Menter SST, Menter BSL con una chiusura a EARSM e il modello a singola equazione di Spalart-Allmaras. I risultati hanno mostrato un generale miglioramento con la nuova funzione di parete nella risoluzione del flusso insieme a vincoli di griglia meno stringenti, convergenza dei residui più veloce e bassa, ed una generale riduzione dei tempi computazionali.

Parole chiave

Funzioni di Parete, Flussi a Parete, M-Edge, Fluidodinamica Computazionale, Strato Limite Turbolento, Modelli di Turbolenza.

Contents

List of Figures

List of Tables

List of Symbols

δ	Boundary layer thickness	m
δ_ν	Viscous length scale ($= \nu/u_\tau$)	m
κ	von Karman constant (=0.41)	
$\mathcal{P}$	Production of turbulent kinetic energy	m^2/s^3
$\mathcal{V}$	Volume of a control volume of the dual grid	
μ	Molecular dynamic viscosity	$Pa \cdot s$
μ_T	Turbulent dynamic viscosity	$Pa \cdot s$
ν	Molecular kinematic viscosity ($= \mu/\rho$)	m^2/s
ν_T	Turbulent kinematic viscosity ($= \mu_T/\rho$)	m^2/s
ω	Turbulent frequency	s^{-1}
Ω_{ij}	Mean rate-of-rotation tensor	s^{-1}
ρ	Density	kg/m^3
τ_w	Wall shear stress	Pa
ε	Dissipation rate of turbulent kinetic energy	m^2/s^3
a_{ij}	Reynolds stresses anisotropic tensor	
C_d	Drag coefficient	
C_f	Skin friction coefficient	
C_l	Lift coefficient	
C_p	Pressure coefficient	
k	Turbulent kinetic energy	m^2/s^2
q	Primitive variables in RANS equations	

S	Strain rate ($= \sqrt{2S_{ij}S_{ij}}$)	s^{-1}
S_{ij}	Mean rate-of-strain train tensor	s^{-1}
s_{ij}	Fluctuation rate-of-strain tensor	s^{-1}
u^+	Scaled wall-tangent velocity ($= U/u_\tau$)	
u_τ	Friction velocity ($= \sqrt{\tau_w/\rho}$)	m/s
y^+	Wall unit, scaled wall-normal coordinate ($= y/\delta_\nu$)	

Acronyms and abbreviations

CFD Computational Fluid Dynamics

EARSM Menter BSL + Wallin & Johansson's EARMS closure

SA Spalart-Allmaras one-equation

SST Menter SST

Chapter 1

Introduction

1.1 Problem definition

Nowadays Computational Fluid Dynamics (CFD) is a powerful and widely used tool for developing projects that concern flow motion, in very different fields: from designing aircrafts to optimizing race-car performance, from energy-generating plants to buildings' air and temperature conditioning. As an example, a manufactured refinement of a car that before required hundreds of hours of work in a wind tunnel now can be done in few days through CAD design and CFD simulation.

Even if a powerful tool, CFD remains a mere model of reality, a discretized approximation of something that is a continuum. As a model, it has some critical features that can preclude a simulation to give a consistent solution that actually represents the real situation. These features are mainly related to the experiment discretization, i.e. the computational grid, the flow's modelling and the numerical method, how the governing equations are dicretized and solved. Alongside the consistency of the model, a CFD problem setup should be aim-oriented: the accuracy and complexity of the model should correspond to the desired accuracy of the results and to the desired computational cost. In fact, even if the computing power of existing machines and clusters is considerably high, often the computational resources and the time available to obtain results are limited, hence the computational efficiency of the simulation plays a key role. For all this reasons, CFD simulations are always a compromise between computational resources and modelling level. In this perspective wall function boundary conditions find their functionality.

In wall-bounded viscous flows a *boundary layer* develops, a region close to the wall where most the viscous effects are concentrated. This boundary layer is usually very thin compared to the characteristic dimensions of the body and resolving it properly in a CFD simulation is essential to obtain a consistent flow development. Furthermore, numerous experiments have shown that in turbulent boundary layers the near-wall region can be largely subdivided into smaller layers, like shown in Fig. 1.1.

In the viscous sublayer turbulent phenomena are damped by the presence of the surface and the viscous effects predominate over turbulent ones. Furthermore, the variations of the dependent variables are quite steep near a wall boundary. As a consequence, one has to

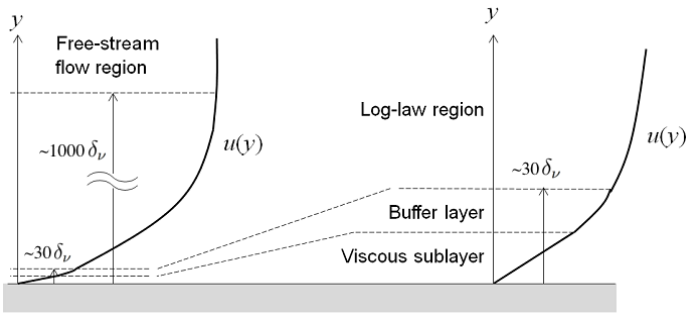


Figure 1.1 – Subdivision of the near-wall region in a boundary layer

employ a method that takes into account the non-linearity of the variables' profile resulting from pressure gradient, mass transfer, transport-property variation. Fully resolving the inner regions of the boundary layer, whose dimensions are in the order of the smaller turbulent structures, require extremely refined computational-grids in the proximity of the wall. This, more than being a severe computational expanse, can lead to numerical problems.

An help comes from the fact that these inner regions are characterized by a universal description. Wall functions are a cheap method that relies on algebraic relations, derived from the universal description of the inner layers by Pradtl's *law of wall*, and avoid the large computational penalty of employing an extremely fine grid that extends all the way to the wall. Most standard wall functions are based on empirically derived profiles of velocity, length scale, shear stress, and turbulent kinetic energy, which are applicable only in very simple near-wall flows and can lead to major errors in complex, non-equilibrium flows, where turbulence creation and destruction rates are far from in balance, and transport effects on all dependent variables may be substantial.

1.2 Background

It must be acknowledged that these limitations were recognized from the earliest appearance of wall functions formulations in the late '60s. Spalding [2] developed an elaborate set of formulae that aimed to account for modifications to the usual log-law formulae caused by pressure gradient and mass transfer through the wall, as well as to circumstances where the wall function only had to account for a portion of the sublayer. The original textbook by Patankar and Spalding [3] also incorporated similar, if somewhat less general, wall functions again taking into account effects of mass transfer and pressure gradient.

Yet, these schemes did not long survive but, like presented by Launder and Spalding [4] in 1973, were instead replaced in CFD software by the conventional logarithmic laws for velocity and temperature, the only improvement to the original logarithmic law of Prandtl being that the friction velocity ($u_\tau \equiv (\tau_w/\rho)^{1/2}$) was replaced by the square-root of turbulent kinetic energy at the near-wall node, $k_P^{1/2}$). This modification showed to be crucial in flows involving separation, stagnation and reattachment, where the wall shear stress vanishes. The

reasons behind this retrograde step were several. Briefly, at the level of fundamental research, wall functions were seen to be an inadequate approach for assessing conjectured transport. Moreover, even for industrial type applications it was recognized that relatively advanced wall functions did not give clearly superior results to a simple log-law formulation.

After almost a decade, there have been attempts at effectively refining the wall-function analysis: Chieng and Launder [5] proposed a wall function in which the near-wall cell was divided into two layers: the viscous sublayer and the fully turbulent region. In the viscous sublayer the shear stress, $-\rho\overline{u'v'}$, was assumed to be zero and the turbulent kinetic energy to vary quadratically with wall distance, while in the fully turbulent region both $-\rho\overline{u'v'}$ and k were assumed to vary linearly. Johnson and Launder [6] modified this treatment by introducing a variable viscous sublayer thickness, caused by changes in the near-wall shear stress distribution; but still the fundamental weakness of the logarithmic velocity and temperature variations was retained. Ciofalo and Collins [7] proposed a new approach making the sublayer thickness a function of the local turbulence intensity. Such local turbulence intensity's definition has two different expressions that switch depending on a "local" Reynolds number, extending the applicability of the approach to when the near-wall node was located in the buffer layer.

More recent and elaborated strategies tried to overcome the limitations related to the hypothesis of the semi-logarithmic distribution of velocity and temperature in the near-wall cell. Rung [8] in 1999 formulated a universal wall function which provided an asymptotic matching of the log-layer and the viscous sublayer, not requiring specific grid resolution properties in the near-wall region. Rung's formulation consisted in a *low-Re modification* of the traditional wall function, by means of Taylor series expansion for the mean-momentum and turbulence properties. In the work presented in 2004, Craft et al. [9] describe a wall treatment based on an efficient one-dimensional numerical integration of simplified low-Re model equations. It implements two grids: a primary grid equivalent to that used with standard high-Re models and conventional wall functions (with a large near-wall cell); and a wall-function grid which is used to calculate source terms, similar to those approximated in standard treatments. The essential difference from earlier approaches is that instead of assuming profiles of velocity and length scale, the profiles of the mean flow and turbulence parameters across this near-wall control volume are obtained from solving simplified boundary-layer-type transport equations.

In [10] Morgenweck validates against each other, and against experimental data, two extended wall-function boundary condition: one is Rung's formulation, based on Taylor series expansion, and the other one is based on exponential transition. The two are implemented in the M-Edge solver and tested on a standard flat-plate case with different grid refinements. Results showed that both have good performances in terms of accuracy but the exponential-transition based one could also handle zero-velocity gradient normal to the wall.

1.3 Purpose of the Present Project

The extended wall-function boundary condition presented by Morgenweck in [10] represents the state-of-art implementation in the M-Edge solver, for $k - \omega$ models. The focus of this master thesis work is the development and validation of a generalized formulation to substitute the current one, improving the performances of the solver. In fact Morgenweck's formulation has presented some lack of accuracy in complex and more general flows, like boundary-layer separation, with wrong predictions of physical behaviour, as well as some problems in numerical convergence and stability. As M-Edge is a solver designed mainly for computational aerodynamics, the purpose is to formulate an efficient but simple wall function that showcases good performances in flows of interest, i.e. external aerodynamic flows. The new formulation of wall-function boundary condition is implemented in the solver's source code and tested on different study cases, in order to understand the differences compared to the previous formulation. Intended improvements for the new formulation are more easy and agile grid requirements, better resolution of strong pressure gradient flows and a formulation for Spalart-Allmaras models. Alongside with these main objectives, working on the presented project aims to a deeper understanding of how building a quality mesh grid and how that, together with the principal input parameters, influences the result and the efficiency of a CFD simulation. To create computational grids will be used Pointwise, a mesh generator software for CFD.

In Chapter 2 the background theory is presented. An introduction on turbulent wall flows is followed by the description of the three different turbulence models employed to test the new wall function. Previous and improved formulation of wall function are presented, outlining the differences in the expressions and in the implementation.

Chapter 3 talks about the methods and the methodology of the work. Specifics on M-Edge, Pointwise and the meshing process are described. Then the research process is briefly described, with explanations on how results are obtained and analyzed, followed by the justification of the methods adopted.

In Chapter 4 the simulations' results are presented and discussed. The validation of the new formulation will start from simple and near-ideal cases (zero pressure gradient flat plate) and advance to always more complex flow cases and geometries (full 3D general fighter). In such a way the first results can be compared to literature reports and then work as a support to more advanced cases' validation.

Chapter 2

Wall Functions and Related Theory

2.1 Turbulent Flows

Turbulent flows are very common to observe in everyday life, whether the smoke from a chimney, a waterfall or a contrail left behind by an airplane. In engineering applications are prevalent but less easily seen; in general the flows around vehicles, e.g. airplanes, cars and ships, are turbulent. Even from a first observation, the characteristics of a turbulent flow are evident: unsteady, irregular and seemingly chaotic and random. Mathematically, this implies that the fluid velocity field varies irregularly and significantly in both space and time.

For fluid flows, both laminar and turbulent, the governing laws are embodied in the Navier–Stokes equations. Even if relatively simple, these equations describe accurately and in detail fluid flows, in all their variety and complexity. However, in the context of turbulent flows, their power is also their weakness: the equations describe every detail of the turbulent velocity field from the largest to the smallest length and time scales. The amount of information contained in the velocity field is vast, and as a consequence (in general) the direct approach of solving the Navier–Stokes equations is out of reach with the present computational resources.

The direct approach of solving the Navier–Stokes equations for turbulent flows is called direct numerical simulation (DNS). It has been proved that the number of operations required by a DNS simulation grows with a parameter that characterizes every fluid flow, the *Reynolds number*. Such quantity represents the ratio of inertial forces to viscous forces within the fluid and is defined as

$$Re = \frac{UL}{\nu} \tag{2.1}$$

where U and L are characteristic velocity and length of the flow and ν is the kinematic viscosity of the fluid. While DNS is intractable for the high-Reynolds-number flows of practical interest, it is nevertheless a powerful research tool for investigating simple turbulent flows at moderate Reynolds numbers.

For the high-Reynolds-number flows that are prevalent in industrial applications, the alternative is to pursue a statistical approach [1]. That is, to describe the turbulent flow,

not in terms of the velocity $\boldsymbol{U}$, but in terms of some statistics, the mean velocity field $\overline{\boldsymbol{U}}$ and its fluctuation $\boldsymbol{u'}$ of the same

$$\boldsymbol{U}(x,t) = \overline{\boldsymbol{U}}(x,t) + \boldsymbol{u'}(x,t) \tag{2.2}$$

also known as the *Reynolds decomposition*. For the sake of simplicity, from now on the overline for the mean velocity will be omitted, unless ambiguities make it necessary. Applying such decomposition to the Navier-Stokes equation, the *Reynolds averaged Navier-Stokes equations (RANS)* are obtained. This set of equations based on the statistic model is tractable, because statistical fields vary smoothly (if at all) in position and time. Looking at the momentum equation, or *Reynolds equation*

$$\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}\left[\nu\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) - \overline{u_i' u_j'}\right] \tag{2.3}$$

a new term appeared, the *turbulent*, or *Reynolds stresses* given by

$$-\rho\overline{u_i' u_j'}. \tag{2.4}$$

This term is an additional unknown in the set of equation, resulting in an unclosed problem; hence one or more additional equation are necessary to model it and close the system. Often, even if it's not fully correct, for the sake of simplicity the tensor $\overline{u_i' u_j'}$ is referred as Reynolds stresses, of which half the trace corresponds to the *turbulent kinetic energy*

$$k \equiv \frac{1}{2}\overline{u_i' u_i'}. \tag{2.5}$$

It is useful to also introduce the normalized *Reynolds stresses anisotropic tensor*

$$a_{ij} \equiv \frac{\overline{u_i' u_i'}}{k} - \frac{2}{3}\delta_{ij} \tag{2.6}$$

which consists in the deviatoric part of $\overline{u_i' u_j'}$. In the stress tensor, it is only the anisotropic part a_{ij} that is effective in transporting momentum. Two relevant quantities that play a role in the transport equation of k are the *production*, $\mathcal{P}$, and the *dissipation*, ε, of turbulent kinetic energy. Such terms read as

$$\mathcal{P} \equiv -\overline{u_i' u_j'}\frac{\partial Ui}{\partial x_j} = -k a_{ij} S_{ij}, \tag{2.7}$$

$$\varepsilon \equiv 2\nu\overline{s_{ij} s_{ij}} \tag{2.8}$$

where S_{ij} and s_{ij} sre respectively the mean and fluctuating strain rate tensor

$$S_{ij} = \frac{1}{2}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right), \qquad s_{ij} = \frac{1}{2}\left(\frac{\partial u_i'}{\partial x_j} + \frac{\partial u_j'}{\partial x_i}\right). \tag{2.9}$$

2.1.1 Wall Flows

In industrial applications, most turbulent flows are bounded by one or more solid surfaces; internal flows, such as flow through a duct or a pipe, and external flows, such as the flow around cars and aircraft, are examples of this category of flows. These *wall flows* are all characterized by the no-slip condition induced by the presence of the solid surface: all the velocity components at wall are equal to zero, usually creating a steep gradient in the tangential velocity profile normal to the wall. When studying this flows, central issues are the form of the mean velocity profiles and of the shear stress exerted by the fluid on the wall, strictly related to the velocity gradient.

For the interest of this work, a schematic boundary layer flow will be treated in the following. Neverthless, as will be assessed later on, the description of the inner regions of the boundary layer is universal and in common with other wall flows, such as the channel flow.

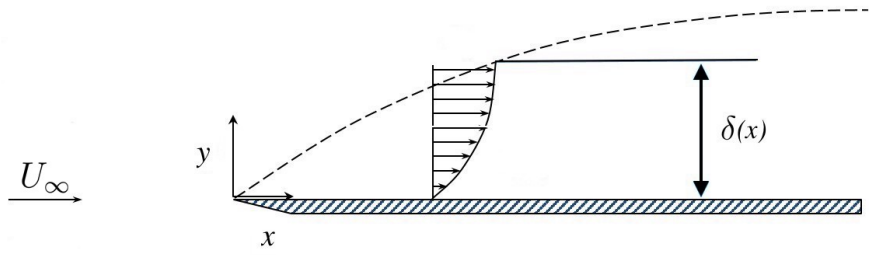


Figure 2.1 – Sketch of a flat-plate boundary layer case.

As sketched in Figure 2.1, is at $y{=}0$, with the leading edge at $x\ {=}0, y{=}\ 0$. If a characteristic velocity value can be represented by the free-stream velocity U_∞, a global length characteristic is the boundary layer thickness $\delta(x)$, that increases along the x-direction as the boundary layer develops along the plate. Assuming the flow fully-developed and two dimensional, together with the boundary-layer approximation the mean momentum equation 2.3 becomes

$$U\frac{\partial U}{\partial x} + V\frac{\partial U}{\partial y} = -\frac{1}{\rho}\frac{dp}{dx} + \frac{1}{\rho}\frac{\partial \tau}{\partial y}, \tag{2.10}$$

where $\tau(x, y)$ is the *total shear stress*

$$\tau = \rho\nu\frac{\partial U}{\partial y} - \rho\overline{u'v'} \tag{2.11}$$

sum of the viscous stress and the Reynolds stress. At wall, the no-slip condition dictates that also the Reynolds stresses are zero. Consequently the shear stress at wall is due entirely to the viscous contribution,

$$\tau_w \equiv \rho\nu \left(\frac{\partial U}{\partial y}\right)_{y=0}, \tag{2.12}$$

where τ_w is called *wall shear stress* and, normalized with a reference velocity, gives the *skin friction coefficient*

$$C_f \equiv \frac{\tau_w}{\frac{1}{2}\rho U_\infty^2}. \tag{2.13}$$

It is evident that, close to the wall, the viscosity ν and the wall shear stress τ_w are important parameters. From these quantities (and ρ) are defined *viscous (or inner) scales* that are the appropriate velocity scales and lengthscales in the near-wall region. These are the *friction velocity* u_τ and the *viscous lengthscale* δ_ν

$$u_\tau \equiv \sqrt{\frac{\tau_w}{\rho}}, \qquad \delta_\nu \equiv \nu\sqrt{\frac{\rho}{\tau_w}} = \frac{\nu}{u_\tau}. \tag{2.14}$$

The distance from the wall, i.e. the wall-normal coordinate, measured in viscous lengths (or *wall units*) is denoted by

$$y^+ \equiv \frac{y}{\delta_\nu} = \frac{u_\tau y}{\nu}. \tag{2.15}$$

Notice that y^+ is similar to a local Reynolds number, so its magnitude can be expected to determine the relative importance of viscous and turbulent processes [1]. In support of this supposition, Fig. 2.2 shows the fractional contributions to the total stress from the viscous and Reynolds stresses in the near-wall region of a channel flow (similar results are also observed in boundary-layer flows).

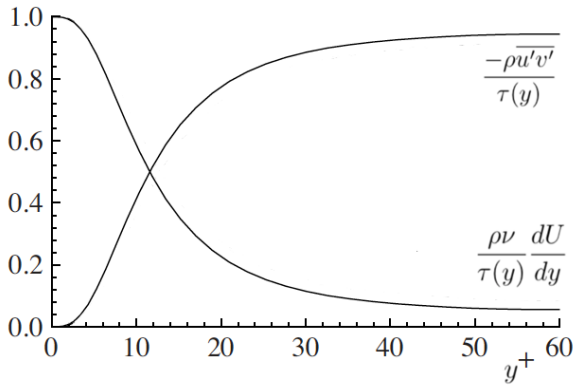


Figure 2.2 – Profiles of the fractional contributions of the viscous and Reynolds stresses to the total stress. DNS data:solid lines, Re = 13,750.

Different regions, or layers, in the near-wall flow are defined on the basis of y^+. In the *viscous wall region* $y^+<50$, there is a direct effect of molecular viscosity on the shear stress;

whereas, conversely, in the *outer layer* $y^+>50$ the direct effect of viscosity is negligible. Within the viscous wall region, in the *viscous sublayer* $y^+<5$, the Reynolds shear stress is negligible compared with the viscous stress.

Considering a zero pressure-gradient case, the wall-normal velocity gradient is found to be fully determinable if the quantities ρ, ν, δ and τ_w are known. Without any loss of generality, it can be rewritten as

$$\frac{\partial U}{\partial y} = \frac{u_\tau}{y}\Phi\left(\frac{y}{\delta_\nu}, \frac{y}{\delta}\right). \tag{2.16}$$

In 1925 Prandtl postulated that there's an *inner layer* close to the wall ($y \ll \delta$) where the mean velocity profile is independent of δ and U_∞, hence the universal function Φ depends only on y/δ_ν. Integrating and applying the viscous scaling to Eq. 2.16, the *law of the wall* is obtained

$$u^+ = f_w(y^+) \tag{2.17}$$

where

$$u^+ \equiv \frac{U}{u_\tau}, \qquad f_w(y^+) = \int_0^{y^+} \frac{1}{y^*}\Phi(y^*)dy^*. \tag{2.18}$$

It is important to stress the relevance of the fact that, according to Prandtl's hypothesis, u^+ depends solely on y^+ for $y \ll \delta$ and there's adbundant experimental verification that the function f_W is universal for wall flows.

In the limit of $y^+ \to 0$, the no-slip boundary condition leads to

$$f_w = y^+ + \mathcal{O}(y^{+2}). \tag{2.19}$$

The departures from the linear relation $u^+ = y^+$ are negligible in the viscous sublayer ($y^+<5$), but are significant (greater than 25%) for $y^+>12$.

As shown in Fig. 2.2, at large values of y^+ it can be supposed that the viscosity has little effect. Hence, in Eq. 2.16, the dependence of $\Phi(y/\delta_\nu)$ on ν (through δ_ν) vanishes, so that Φ adopts a constant value

$$\Phi(y^+) = \frac{1}{\kappa} \qquad \text{for } y^+ \gg 1,\ y \ll \delta. \tag{2.20}$$

In this region Eq. 2.17 assumes the form

$$u^+ = \frac{1}{\kappa}\ln y^+ + B \tag{2.21}$$

which takes the name of *log law*, where B is a constant and κ is the von Karman constant. In the literature, such constants generally assume the values of

$$B = 5.2 \qquad \text{and} \qquad \kappa = 0.41. \tag{2.22}$$

Looking at Fig. 2.3, it's possible to observe how the scaled velocity profiles have an excellent agreement with the log law for $y^+>30$, that takes the name of *log-law region* (or *log layer*). The region between the viscous sublayer ($y^+<5$) and the log-law region ($y^+>30$) is called the buffer layer. It is the transition region between the viscosity-dominated and the turbulence-dominated parts of the flow and an expression for the law of the wall in this region is not as direct as the ones discussed previously. In 1956 van Driest formulated an extended expression for the law of the wall, which is not repeated here and can be found in Chapter 7 of [1]. Van Driest's law of wall recovers the linear relation and the log law in the respective regions and also provides an excellent representation of the data, as Fig. 2.3 showcases.

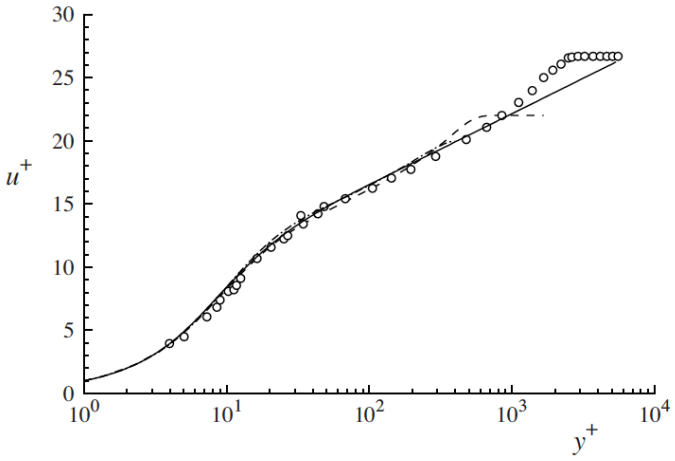


Figure 2.3 – Mean velocity profiles in wall units. Circles, boundary-layer experiments, Re_θ = 8,000; dashed line, boundary-layer DNS Re_θ = 1,410; dot–dashed line, channel flow DNS, Re = 13,750; solid line, van Driest's law of the wall. Figure taken from [1].

2.1.2 Turbulence Modelling

A large variety of Reynolds stresses models are the *eddy viscosity models* which are based on the *turbulent-viscosity hypothesis*. The hypothesis, introduced by Boussinesq in 1877, consists in the idea that in turbulent flows the momentum mixing is mainly governed by large energetic turbulence eddies, in a similar way as molecular mixing, and reads as

$$a_{ij} = -\frac{\nu_T}{k}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right) = -2\frac{\nu_T}{k}S_{ij} \tag{2.23}$$

where ν_T is the *eddy viscosity* (also called *turbulent viscosity*).

A relevant limit of the eddy viscosity models is the alignment of the tensor a_{ij} with the mean rate-of-strain tensor S_{ij}. Both the tensor a_{ij} and S_{ij} are deviatoric and have five independent components, a_{ij} due to the definition and S_{ij} due to divergence free incompressible flows; such five components, according to the Boussinesq hypothesis (Eq. 2.23), are related to each other through the scalar ν_T. Usually this is not the case in real situations and even in simple shear flows the deviation from this alignment is rather large.

Applying the Boussinesq hypothesis to the mean-momentum equation (i.e Eq. 2.23 into Eq. 2.3)

$$\frac{\partial U_i}{\partial t} + U_j \frac{\partial U_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial}{\partial x_i}\left(p + \frac{2}{3}\rho k\right) + \frac{\partial}{\partial x_j}\left[\nu_{eff}\left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i}\right)\right] \tag{2.24}$$

where

$$\nu_{eff}(\boldsymbol{x}, t) = \nu + \nu_T(\boldsymbol{x}, t) \tag{2.25}$$

is the *effective viscosity* and the isotropic part of the Reynolds stress tensor is incorporated in the pressure term.

The eddy viscosity cannot be assumed to be constant but rather governed by the length scale (Λ) and velocity scale (V) of the large energetic eddies. Dimensional arguments then give the following relation

$$\nu_T \sim \Lambda V. \tag{2.26}$$

The modelling of the Reynolds stress tensor with six components in general three-dimensional flows is then reduced to model the eddy viscosity, or equivalent, to model the large eddy length and velocity scales. This is a large reduction in complexity which is attractive for implementation and use in CFD of general flows. The approaches to specify (Λ) and (V) are many and vary depending on the type of eddy viscosity model. Such models can be classified in:

- **Algebraic models**, or zero equation models, are the simplest models. Length Λ and velocity V scales are related to some characteristic velocity and geometry of the mean flow. As said, this models are very simple but hard to apply to a broad range of flow cases. They are characterized in fact by a considerable level of empirism when choosing the characteristic scales, especially if a complex and new problem is approached. For standard and extensively studied flows, such as attached boundary layers, the specification of the scales is well established and algebraic models represent a cheap and efficient choice.

- **One-equation models** typically consist in solving a transport equation for the turbulent kinetic energy, k, or for the eddy viscosity, ν_T. Similarly to algebraic models, one-equation models also need to some extent a specification from global conditions but it's more general than in zero-equation models. One-equation models are well suited for attached boundary layers and other thin shear flows, but cannot be expected to work very well in complex flows. They are widely optimized and used in the aeronautical industry.

- **Two-equation models** consist in solving the transport equation for two different turbulent variables that usually are the turbulent kinetic energy, k, its dissipation rate, ε, or the turbulent frequency ω. From these two quantities a lengthscale, a time scale and

consequently a velocity scale can be formed, making the models *complete* and flow-dependent specifications generally not required. However, the models may contain additional corrections that might depend on global and flow-related quantities, e.g. the wall distance. Two-equation models are the most commonly implemented models in CFD, with proven good performances for attached boundary layers and other thin shear flows, being also reasonably general in other flows.

During the development of the presented project, three different eddy viscosity turbulent models were implemented and are now presented in the following.

Spalart-Allmaras one-equation model

First presented in 1992, Spalart and Allmaras [11] described a one-equation model developed for aerodynamic applications, in which a single model transport equation is solved for the quantity $\tilde{\nu}$, that is equivalent to the turbulent viscosity ν_T. The interesting feature of $\tilde{\nu}$ is that it equals ν_T in the whole boundary layer, except within the viscous region, where $\tilde{\nu}$ is linear with the wall distance and ν_T is not. The two quantities are related by

$$\nu_T = \tilde{\nu} f_{v1} \tag{2.27}$$

where

$$f_{v1} = \frac{\tilde{\chi}^3}{\tilde{\chi}^3 + C_{v3}^3}, \qquad \tilde{\chi} \equiv \frac{\tilde{\nu}}{\nu}. \tag{2.28}$$

The transport equation for $\tilde{\nu}$ reads as

$$\frac{D\tilde{\nu}}{Dt} = C_{b1}\tilde{S}\tilde{\nu} + \frac{1}{\sigma}\left[\frac{\partial}{\partial x_j}\left((\nu + \tilde{\nu})\frac{\partial \tilde{\nu}}{\partial x_j}\right) + C_{b2}\frac{\partial \tilde{\nu}}{\partial x_j}\frac{\partial \tilde{\nu}}{\partial x_j}\right] - C_{w1} f_w \left[\frac{\tilde{\nu}}{d}\right]^2 \tag{2.29}$$

and the terms on the right hand side are, in order, the production term, the diffusive term and the destruction term. The complete formulation of the model includes an additional set of terms that provides control over laminar regions of shear layers - keeps the flow laminar where desired, obtains transition where desired - but have been omitted here for the sake of simplicity. Further details on the model, the calibration and empirical constants, C, and functions, f, can be found in the reference [11].

The model is designed for aeronautical flows and represents a model computationally easier than two-equation models but more complete and accurate than k-based one-equation models. Since turbulence is characterised by two scales and the model only solves for one property additional information is needed. The Spalart-Allmaras model uses the wall distance, d in Eq.2.29, that would be active through the complete boundary layer. The model is integrated all the way to the wall which requires a good grid resolution normal to the wall ($y^+ \sim 1$).

Menter BSL and SST $k-\omega$ models

The $k-\omega$ models are extensively used in CFD, especially for high Reynolds number flows around vehicles. In the largest part of $k-\omega$ models, ω is defined as

$$\omega \equiv \frac{\varepsilon}{C_\mu k} \tag{2.30}$$

where the standard value of the model coefficient C_μ is 0.09, and the following expression for the eddy viscosity is

$$\nu_T = \frac{k}{\omega}. \tag{2.31}$$

Dimensionally, $\omega \sim \varepsilon/k$ is a frequency and can be interpreted as the frequency of the large scale eddies, hence the name of *turbulent frequency*. The standard formulation of $k-\omega$ model, as described by Wilcox in [12], for boundary layer flows, is superior to standard $k-\varepsilon$ model both in its treatment of the viscous near-wall region and in the accounting for the effects of streamwise pressure gradients. The major problem with the standard $k-\omega$ model is the treatment of turbulence interfaces at e.g. the boundary layer edge, leading to unphysical sensitivity to free-stream values of k and ω. Such problem does not affect the $k-\varepsilon$ model and Menter proposed in [13] a two-equation model designed to yield the best behavior of the $k-\varepsilon$ and $k-\omega$ models. This new model, named Baseline (BSL) model, is obtained blending a standard $k-\varepsilon$ (transformed into $k-\omega$ formulation) with the Wilcox $k-\omega$ model, resulting in the two equations

$$\frac{Dk}{Dt} = \mathcal{P} - C_\mu k\omega + \frac{\partial}{\partial x_j}\left(\left(\nu + \frac{\nu_T}{\sigma_k}\right)\frac{\partial k}{\partial x_j}\right), \tag{2.32}$$

$$\frac{D\omega}{Dt} = C_{\omega 1}\frac{\mathcal{P}}{\nu_T} - C_{\omega 2}\omega^2 + \frac{\partial}{\partial x_j}\left(\left(\nu + \frac{\nu_T}{\sigma_\omega}\right)\frac{\partial \omega}{\partial x_j}\right) + 2\left(1-F_1\right)\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}. \tag{2.33}$$

The last term of Eq. 2.33 is a cross-diffusion term introduced with the blending and the *blending function* F_1 is designed to be one in the sublayer and logarithmic region of the boundary layer, activating the $k-\omega$ model, and to gradually switch to zero in the wake region, $k-\varepsilon$ model activated.

In a further improvement the definition of the eddy viscosity is modified to account for the transport of the principal turbulent shear stress $-\overline{u_1' u_2'}$. The resulting model is called the Shear-Stress Transport (SST) model. According to the Bradshaw assumption, which states that the shear stress in a boundary layer is proportional to the turbulent kinetic energy k, it follows

$$-a_{12} = -\frac{\overline{u_1' u_2'}}{k} = a_1, \qquad \text{with } a_1 = 0.3 \tag{2.34}$$

while with the Boussinesq assumption and relations 2.30 and 2.31 it reads

$$-a_{12} = 2C_\mu \frac{k}{\varepsilon} S_{12} = C_\mu \frac{k}{\varepsilon} S, \tag{2.35}$$

where $S = \sqrt{2S_{ij}S_{ij}}$ is a measure of the strain rate. The latter leads to a modelled production proportional to S^2, that this way can be severally overpredicted in cases with large strain rate, like adverse pressure gradient flows. The solution that Menter proposes in [13] is to limit the anisotropy to

$$-a_{12} = \min\left(C_\mu \frac{k}{\varepsilon} S, a_1\right) \tag{2.36}$$

limitation that can be extended to the eddy viscosity as

$$\nu_T = \frac{a_1 k}{\max(a_1\omega, S)}. \tag{2.37}$$

Results in Menter's work [13] showed that this limitation is vital in the resolution of strongly adverse pressure gradient flows, accurately predicting pressure-induced separation and the resulting viscous-inviscid interaction.

Explicit Algebraic Reynolds Stress Models (EARSM)

The eddy-viscosity-based two-equation models have been dominating in the context of industrial CFD so far, but demonstrated poor accuracy in increasingly challenging flow situations, which include onset of separation, highly curved flows, rapidly rotating flows etc. The weakness of Boussinesq hypothesis is the supposed linearity between the turbulent anisotropic tensor and the strain-rate tensor, which results in a crude modelling of the production, insensitive to system rotation. The level of Differential RSM includes without doubt much more of the flow physics in a natural way, but the implementation in codes for design work of complex industrial flows has proven to still be a considerable challenge. In this sense, EARSM have demonstrated to be a good compromise. In algebraic Reynolds stress models the aim is to remove the linear hypothesis and replace it with a more general, still local and algebraic, anisotropy relation. Furthermore, it is derived as an approximation of the transport equation for a_{ij}, and thereby inherits properties from the DRSM level in a natural way.

The EARSM by Wallin & Johansson [14] is a rational approximation of a full Reynolds stress transport model in the weak equilibrium limit where the Reynolds stress anisotropy may be considered constant in time and space. The Reynolds stress tensor is explicitly expressed in terms of the velocity gradient and the turbulence scales. The EARSM may be written in a common way where the anisotropy tensor a_{ij} is written in terms of the strain- and rotation rate tensors S_{ij} and Ω_{ij} as

$$a_{ij} = \sum_{\lambda=1}^{10} \beta_\lambda T_{ij}^{(\lambda)} \tag{2.38}$$

where the β coefficients and the and the T_{ij} are respectively functions of the five invariants of-

and powers of tensor combinations of S_{ij} and Ω_{ij}. In Wallin & Johansson's formulation five terms are retained, but in 2D flows only β_1 and β_4 are different from zero and their expression can be found in [14]. This constitutive relation for the Reynolds stress tensor is implemented in the M-Edge solver being coupled to different $k-\omega$ models; in the presented work it is used as a closure of the Menter BSL $k-\omega$ model.

2.2 Wall Functions

The idea behind the wall-function approach is to apply boundary conditions (based on law-of-the-wall relations) at the first inner grid pint, some distance away from the boundary wall, so that the turbulence-model equations are not solved close to the wall (i.e. between the wall and the location at which boundary conditions are applied). Standard wall functions, like the one discussed in [4], rely strictly on semi-logarithmic relations derived from Eq. 2.21 and hence require the first inner point to be located in a region where the log law is valid and accurate. Traditionally this region is indicated as $30<y^+<300$ but, depending on the flow case considered, the overall solution can be very sensitive to the choice of the node's location. Extended wall functions, like Morgenweck's in [10], overcome this problem with formulations that merge the velocity-profile relation for the viscous sublayer and for the log-law region, in such a way that also the buffer layer is properly resolved. This results in much less strict requirements for the first inner node and an overall solution less sensitive to those requirements. Below the previous formulation by Morgenweck is briefly introduced and described. More focus will be put subsequently on the improved formulation, describing its features and differences with the previous one.

2.2.1 Previous Formulation

The expression presented by Morgenweck in [10] provides a transition from the log-relation to the linear relation by means of exponential functions. This class of function is known to perform reasonable and smooth transitions between regions of different influence. Such modification reads as

$$u^+ = \left(1 - e^{-ay^+}\right)\left(\frac{\ln y^+}{\kappa} + B\right) \tag{2.39}$$

The exponential transition is merely an empirical approach, with no analytic foundation. As y^+ goes to zero, the transition factor bends the logarithmic curve to the linear one but for $y^+<1$ Eq. 2.39 becomes negative. Consequently the range of validity is limited using a switch. The full boundary condition for u^+ reads as

$$u^+ = \begin{cases} y^+ & \varphi \leq \varphi_c = y_c^{+2} \\ \left(1 - e^{-ay^+}\right)\left(\frac{\ln y^+}{\kappa} + B\right) & \varphi > \varphi_c = y_c^{+2} \end{cases} \tag{2.40}$$

where a=0.1569 and y_c^+=4.0369532797, derived to ensure the continuity between the two curves. The values of κ and B are the same as in Eq. 2.21. Deriving the above expression along the wall-normal scaled coordinate y^+, the boundary condition for the velocity derivative is obtained

$$\frac{\partial u^+}{\partial y^+} = \begin{cases} 1 & \varphi \leq \varphi_c = y_c^{+2} \\ \left(1 - e^{-ay^+}\right)\frac{1}{\kappa y^+} + ae^{-ay^+}\left(\frac{\ln y^+}{\kappa} + B\right) & \varphi > \varphi_c = y_c^{+2} \end{cases} \tag{2.41}$$

The quantity φ is defined as

$$\varphi = u^+ y^+ = \frac{\rho U y}{\mu} \tag{2.42}$$

which has the relevant characteristic of being independent of u_τ, as shown by the last term of the equivalence.

Buondary conditions for the turbulent quantities are

$$\frac{\partial k}{\partial y} = 0, \qquad \mathcal{P} = \frac{C_\mu^{0.25} k^{0.5} u_\tau^2}{\kappa y}, \qquad \omega = \omega_{loRe} + \omega_{hiRe} \tag{2.43}$$

where

$$\omega_{loRe} = \frac{6\mu}{\rho \beta y^2} \qquad \text{and} \qquad \omega_{hiRe} = \frac{u_\tau}{C_\mu^{0.5} \kappa y} \tag{2.44}$$

which are derived simplifying Eq. 2.32 respectively for low-Re (viscous sublayer) and high-Re (log-law region) conditions. The Neumann boundary condition for k is simple but very effective as, compared to setting it equal to zero at wall, works for the linear law as well as for the log law [10].

2.2.2 Improved Formulation

This section presents the expressions of the improved formulation and how it is implemented in the solver routine. The content is based on [15]. The new extended wall law reads as

$$u^+ = \begin{cases} \frac{1}{a} \sin a y^+ & \varphi \leq \varphi_c \\ \left(1 - c_1' e^{-y^+/c_2}\right) \left(\frac{\ln y^+}{\kappa} + B\right) & \varphi > \varphi_c \end{cases} \tag{2.45}$$

where now $\varphi_c = (y_c^+/a) \sin a y_c^+$. From this modification, it follows that the expression for the velocity gradient becomes

$$\frac{\partial u^+}{\partial y^+} = \begin{cases} \cos a y^+ & \varphi \leq \varphi_c \\ \left(1 - c_1' e^{-y^+/c_2}\right) \frac{1}{\kappa y^+} + \frac{c_1'}{c_2} e^{-y^+/c_2} \left(\frac{\ln y^+}{\kappa} + B\right) & \varphi > \varphi_c \end{cases} \tag{2.46}$$

The c_1' and c_2 coefficients are derived imposing the continuity of the expression and its derivative at the switch ($\varphi = \varphi_c$), resulting in

$$c_1 = 1 - \frac{\frac{1}{a} \sin a y_c^+}{\frac{\ln y_c^+}{\kappa} + B}, \qquad c_2 = \frac{c_1}{a(1 - c_1)} \frac{\sin a y_c^+}{\cos a y_c^+ - \frac{1 - c_1}{\kappa y_c^+}}, \qquad c_1' = c_1 e^{-y_c^+/c_2}. \tag{2.47}$$

It is evident how a and y_c^+ are the parameters that tune the new extended law. The tune of the previous formulation was a=0.1569 and y_c^+=4.0369532797 but it is now modified to better fit low-Re skin friction for all possible resolutions. Different sets of a and y_c^+ are chosen and

implemented for $k-\omega$ and S-A models, see Fig. 2.4.

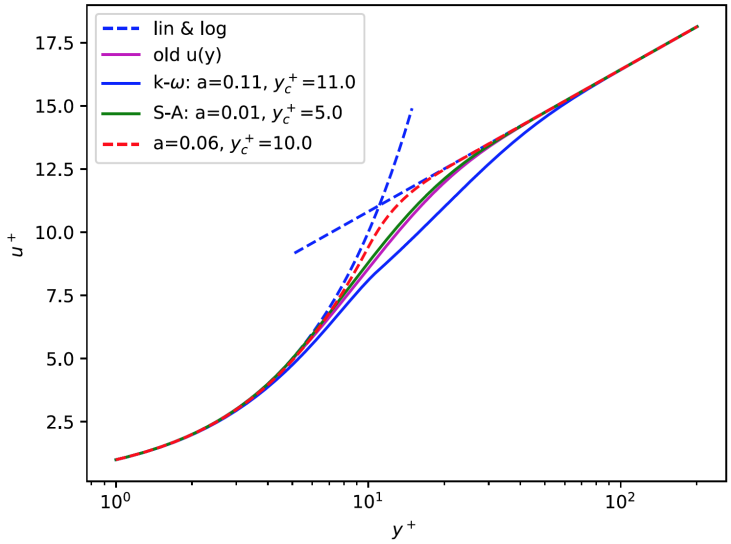


Figure 2.4 – Extended log-law velocity profiles with different a and y_c^+ tunes.

Moreover, simulations on a zero pressure-gradient flat plate showed that the old tune for $k-\omega$ models leads to a relevant instability in the skin friction profile when the boundary-condition node is at the switching point (y_c^+=4.0369532797). In Fig. 2.5 it is possible to observe how the new tune presents a smoother transition across the switch point, compared to old one.

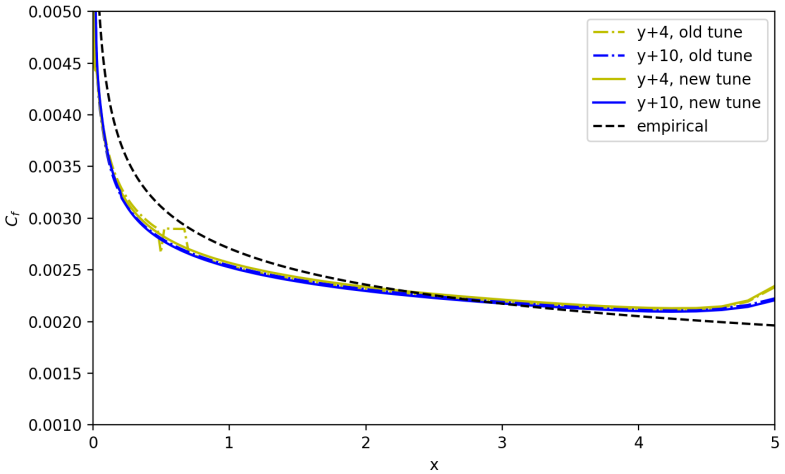


Figure 2.5 – Skin-friction coefficient profiles along a zero pressure-gradient flat plate, with first node at different y^+.

$k-\omega$ Models

Just like described for the previous formulation in 2.2.1, boundary conditions for the turbulent quantities are needed to complete the wall treatment. While for the turbulent kinetic energy k the Neumann boundary condition is still retained, the boundary condition for the production term changes. It can be derived starting from Eq. 2.10. In the very near-wall region, where the flow can be assumed parallel, the equation becomes

$$0 = -\frac{1}{\rho}\frac{dp}{dx} + \frac{1}{\rho}\frac{\partial \tau}{\partial y}. \tag{2.48}$$

Recalling the approximation already made in 2.1.1, neglecting the pressure gradient and integrating along the y-direction, it results that the shear stress is constant through the near-wall region where the wall-function boundary condition is applied

$$\rho\nu\frac{\partial U}{\partial y} - \rho\overline{u'v'} = \tau_w = \rho u_\tau^2. \tag{2.49}$$

With this assumption of constant total shear stress and the Boussinesq hypothesis (Eq. 2.23), it is possible to derive the following, which is valid throughout the near-wall region (i.e. from y^+=0 up to the end of the log layer)

$$-\rho\overline{u'v'} = \rho u_\tau^2\left(1 - \frac{\partial u^+}{\partial y^+}\right) \tag{2.50}$$

$$\mu_T = \mu\left(\frac{1}{\frac{\partial u^+}{\partial y^+}} - 1\right). \tag{2.51}$$

At this point it is possible to define the boundary condition for the production. Recalling the definition of $\mathcal{P}$ in Eq. 2.7, it follows

$$\mathcal{P} = -\overline{u'v'}\frac{\partial U}{\partial xy} = \frac{\rho^2 u_\tau^4}{\mu}\left(1 - \frac{\partial u^+}{\partial y^+}\right)\frac{\partial u^+}{\partial y^+} \tag{2.52}$$

The expression for ω is formally the same used in the previous formulation, i.e. the sum of analytical solutions for the viscous and logarithmic inner layers.

Now it is described the routine in which the wall-function boundary condition is implemented. Indices used to point at grid nodes refer to Fig. 2.6. Index 0 points to quantities at the wall cell ($y \in [0; y_1/2]$) and index 1 to quantities at the first cell away from the wall ($y \in [y_1/2; 3y_1/2]$). Index w is used to point at additional quantities stored at the boundary nodes.

The procedure to apply the boundary conditions is articulated in the following steps:

1. Starting from the value of U_1 in the present solution, the wall friction velocity, u_τ, is derived from the u^+ relation 2.45, solving iteratively an implicit equation (described in the next section). The resulting u_τ is the used to compute the shear stress τ_w, which then is set as a flux in the momentum balance of the wall-adjacent cell.

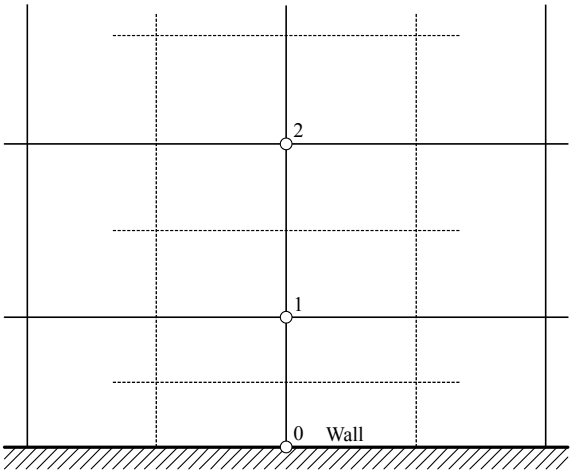


Figure 2.6 – Grid nodes near the wall. Solid line: computational grid; dashed line: dual grid (control volumes).

2. The turbulent frequency, ω, is computed using the relations in 2.44 and strongly set for the wall- and the first inner-node. The y value to use in the relations is set to almost the wall cell center: $y_0 = 0.3y_1$. For the high-Re solution, is proven to be valid the relation $u_\tau = C_\mu^{0.25} k^{0.5}$, hence the expression for ω becomes

$$\omega_{hiRe} = \frac{k^{0.5}}{C_\mu^{0.25} \kappa y}. \tag{2.53}$$

3. Like ω, also the turbulent kinetic-energy production, $\mathcal{P}$, is computed and set strongly for the wall- and first inner-node, by means of the relation

$$\mathcal{P} = \begin{cases} \frac{\rho^2 u_\tau^4}{\mu} \left(1 - \frac{\partial u^+}{\partial y^+}\right) \frac{\partial u^+}{\partial y^+} & y^+ \leq 25 \\ C_\mu \rho k \omega & y^+ > 25 \end{cases} \tag{2.54}$$

Also here, $y_0^+ = 0.3y_1^+$ is used for computing $\partial u^+ / \partial y^+$ from the analytical expression 2.46. Since $k = u_\tau^2 C_\mu^{-0.5}$ and $k = u_\tau C_\mu^{-0.25} k^{0.5}$ subsist respectively for the wall node and the first inner node, thus the outer condition for $\mathcal{P}$ becomes

$$\mathcal{P}_0 = C_\mu^{0.5} \rho_1 u_\tau^2 \omega_1, \qquad \mathcal{P}_1 = C_\mu^{0.75} \rho_1 u_\tau k_1^{0.5} \omega_1. \tag{2.55}$$

4. The turbulent viscosity μ_T is computed and strongly set at wall. It is obtained from the analytical μ_T at the interface $y_{01}^+ = y_1^+/2$ as

$$\mu_{T0} = 2\mu_{T01} - \mu_T 1 \tag{2.56}$$

where μ_{T1} is the present solution. Furthermore, μ_{T0} is limited to minimum zero and μ_{T01}, obtained with relation 2.51, is limited to maximum μ_{T1}.

The substantial differences with the previous formulation's implementation are two. The first concerns the second step. If the relation used for ω is formally the same, the way it is set is different; in the previous formulation it was only set strongly at the wall node, while now it is also set at the first inner node. This improves the numerical stability and convergence of the ω model-equation in the near-wall region. The second and more important difference concerns the first step. As described above, now the shear stress τ_w is computed and directly set as a flux at the interface of wall and first inner control volume, acting directly on the momentum balance of the two volumes. In the previous formulation, the application of this boundary condition was different: τ_w was not directly set but used to compute the velocity at the wall so that the corresponding viscous stress between the wall and the first inner node equals τ_w. If, on one hand, this is a numerically more convenient solution in presence of discontinuities (such as trailing edges) where prescribing fluxes can be problematic [10], on the other hand no flux besides the viscous one is considered, neglecting the turbulence and pressure contribution. Neglecting the turbulent stresses is a poor simplification if the first inner node is positioned at large y^+; the same consideration is valid for the pressure contribution in presence of strong pressure gradients, which can result in an overall poor resolution of pressure forces on the solid surface.

Solution of the implicit wall law

The wall friction velocity u_τ in step 1. is obtained iteratively from the extended wall law. Pivotal in this procedure is the variable φ, previously introduced in 2.2.1, as it is directly computable and monotonic, i.e. each value of φ corresponds to single specific values of u^+ and y^+. The extended relation 2.45 for u^+ can be expressed as

$$u^+(y^+) = u^+ \tag{2.57}$$

and introducing the quantities φ and $\chi = 1/\kappa u^+$, it can be reformulated as the fixed point equation

$$f(\chi;\varphi) = \kappa\chi u^+(\kappa\varphi\chi) - 1. \tag{2.58}$$

The solution $f(\chi;\varphi)$ is obtained by the Newton's method iteration

$$\chi_{new} = \chi - \frac{f(\chi)}{f'(\chi)} \tag{2.59}$$

where

$$f'(\chi) = \kappa u^+(\kappa\varphi\chi) + \kappa^2\varphi\chi u^{+'}(\kappa\varphi\chi). \tag{2.60}$$

Starting from the initial solution $\chi = 1/\kappa\sqrt{\varphi}$, the solution converges rapidly, reaching machine accuracy error in 5 steps, hence the number of iterations in the implementation is set to 5. From the resulting φ_1, the wall friction velocity is then obtained by

$$u_\tau \equiv \sqrt{\frac{\tau_w}{\rho_w}} = \frac{\kappa\varphi\chi\mu_1}{y_1\rho_1} \tag{2.61}$$

remembering that y is the normal distance from the wall.

Spalart-Allmaras models

The turbulent variable that, introduced in 2.1.2, is solved in the Spalart-Allmaras model equation is $\tilde{\nu}$. The extended wall function for $\tilde{\nu}$ is

$$\tilde{\nu}^+ = \kappa y^+. \tag{2.62}$$

Looking at Eq. 2.29, the diffusive term in Spalart-Allmaras model reads

$$\mathcal{D}_{\tilde{\nu}} = \frac{1}{\sigma}\left[\frac{\partial}{\partial x_j}\left(\left(\mu + \rho\tilde{\nu}\right)\frac{\partial \tilde{\nu}}{\partial x_j}\right) + C_{b2}\rho\frac{\partial \tilde{\nu}}{\partial x_j}\frac{\partial \tilde{\nu}}{\partial x_j}\right]. \tag{2.63}$$

Recalling the fact that close to the wall, where the wall-function boundary condition is applied, $\tilde{\nu}$ is linear with the wall distance, the cross diffusion can be expanded and the second derivative of $\tilde{\nu}$ neglected. Hence the near-wall flux is computed as

$$f_{\tilde{\nu}} = \frac{1}{\sigma}\left[\mu + \left(1 + C_{b2}\right)\rho\tilde{\nu}\right]\frac{\partial \tilde{\nu}}{\partial y} \tag{2.64}$$

or, the near-wall face viscosity, used in the calculation of viscous fluxes at wall, reads

$$\mu_f^* = \frac{1}{\sigma}\left[\mu + \left(1 + C_{b2}\right)\rho\tilde{\nu}\right]. \tag{2.65}$$

The production term in the complete form of Eq. 2.29 reads

$$\mathcal{P}_{\tilde{\nu}} = C_{b1}(1 - f_{t2})\tilde{S}\tilde{\nu}, \qquad \text{with} \qquad \tilde{S} \equiv S + \frac{\tilde{\nu}}{\kappa^2 y^2} f_{v2}. \tag{2.66}$$

While at wall $\mathcal{P}_{\tilde{\nu}} = 0$ is used, for the first inner node $\tilde{S}$ needs to be computed. In the model, the function f_{v2} is constructed so that $\tilde{S}$ maintains its log layer behaviour ($\tilde{S} = u_\tau/\kappa y$) all the way to the wall [11]. For first inner nodes at small y^+ this could result in a poor resolution of $\tilde{S}$, hence of the production term, due to the fact that the velocity gradient varies more strongly. To improve this aspect, the extended wall function is used to provide $u^{+'}$; in addition, $\tilde{\nu} = \kappa u_\tau y$ is used to compute $\tilde{S}$ at the first inner node, resulting in

$$\tilde{S} = \frac{\rho u_\tau^2}{\mu}\left(\frac{\partial u^+}{\partial y^+} + \frac{f_{v2}}{\kappa y^+}\right). \tag{2.67}$$

Chapter 3

Methods and Methodology

3.1 M-Edge Flow Solver

The M-Edge code was created in 1997 at former FFA, Aeronautical Research Institute of Sweden. The code is today owned by FOI, the Swedish Defence Research Agency. It is one of the main R&D tools at Saab industries for aeronautical projects, like the fighter aircraft Gripen.
The main features of M-Edge are

- it is a finite-volume flow solver for unstructured grids;
- it runs in parallel for large scale applications, through domain decomposition and MPI communication;
- it uses the in-house FFA data format for input and output files.

M-Edge is a parallelized CFD flow solver system for solving 2D/3D viscous/inviscid, compressible flow problems, for both steady state and time accurate calculations, including manoeuvres and aeroelastic simulations.

The solver employs an edge-based formulation which uses a node-centered finite-volume technique to solve the governing equations [16]. The finite-volume technique requires control volumes, which are non-overlapping and form the dual grid obtained from the control surfaces at the edges of the primary input mesh. The dual grid is generated by the preprocessor as an input to the flow solver. In this pre-simulation phase, the preprocessor is also responsible of fusing the control volumes into coarser grid cells, to be used as multigrid levels, and of partitioning the grid, for parallel computation on multiple processors. In the flow solver, the governing equations are integrated explicitly towards steady state with Runge-Kutta time integration. Convergence is accelerated using multigrid strategies and implicit residual smoothing. Time accurate computations can be performed using a semi-implicit, dual time stepping scheme which implements convergence acceleration through a steady state from inner iteration procedure. In the following sections some main numerical aspects of the code are described, focusing only on the schemes and procedures that are employed for the presented project.

3.1.1 Governing Equations

The governing equations that are computed to solve the flow dynamic are the RANS equations, previously treated in 2.1. In addition to such equations, one or two turbulence-modelling equations are solved to close the system. The finite-volume method solver makes use of the conservative form of the governing equations, which can be obtained integrating over a control volume, Ω, the differential form. The equations to solve hence read

$$\int_{\Omega} \frac{\partial q}{\partial t} dV + \oint_{\partial\Omega} \boldsymbol{F}_I \boldsymbol{n} dS = \oint_{\partial\Omega} \boldsymbol{F}_V \boldsymbol{n} dS + \int_{\Omega} Q dV \tag{3.1}$$

where q are the primitive variables and Q the source terms, that for the different equations denote

$$q = \begin{pmatrix} \rho \\ \rho U \\ \rho V \\ \rho W \\ E \\ \rho Z \end{pmatrix}, \qquad Q = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ \mathcal{P}_Z - \varepsilon_Z \end{pmatrix}. \tag{3.2}$$

$\boldsymbol{F}_I$ and $\boldsymbol{F}_V$ are respectively the the inviscid and viscous fluxes, which are expressed in Cartesian components as

$$\boldsymbol{F} = f_1 e_x^T + f_2 e_y^T + f_3 e_z^T \tag{3.3}$$

with

$$f_{I_i} = \begin{pmatrix} \rho U_i \\ \rho U U_i + p^* \delta_{i1} \\ \rho V U_i + p^* \delta_{i2} \\ \rho W U_i + p^* \delta_{i3} \\ \tau_{ij} U_j \\ \rho Z U_i \end{pmatrix} \qquad f_{V_i} = \begin{pmatrix} 0 \\ \tau_{i1} \\ \tau_{i2} \\ \tau_{i3} \\ \tau_{ij} U_j - q_i + (\mu + \sigma_k \mu_T) \frac{\partial k}{\partial x_i} \\ (\mu + \sigma_Z \mu_T) \frac{\partial Z}{\partial x_i} \end{pmatrix}. \tag{3.4}$$

In the above expressions Z is a general turbulent quantity (e.g. k or ω), $p* = p + 2/3\rho k$ and q_i are the heat fluxes. For a caloric perfect gas, as is assumed in the project's simulations, the total energy E reads

$$E = \frac{p}{\gamma - 1} + \frac{1}{2} \rho \boldsymbol{U}\boldsymbol{U} + \rho k. \tag{3.5}$$

3.1.2 Spatial Discretization

The finite-volume technique requires control volumes surrounding the nodes of the input grid, in which the unknowns are stored. The fluxes are calculated alongside the edges, the gridlines

that connect two nodes. Fig. 3.1 shows how a dual grid (dashed lines) is constructed: the

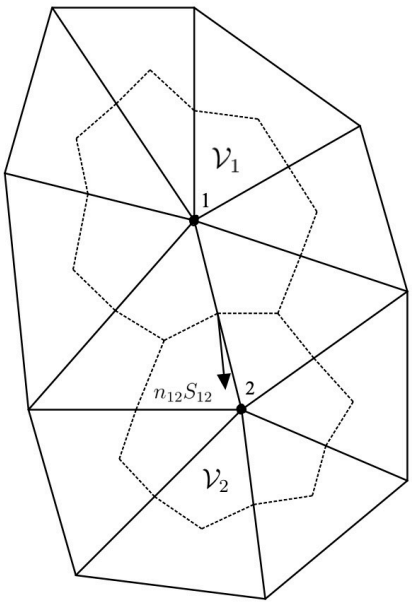


Figure 3.1 – The unstructured input grid (solid lines) and the dual grid (dashed lines) forming the control volumes.

control surfaces around a node are obtained by connecting the midpoint of each edge that depart from a node to the center of mass of the elements that share the same node. The control surfaces of all edges emerging from a node enclose the control volume of a node.
The coordinates of the input grid are provided for each node and the connectivity is supplied in an edge based manner where an edge connects two nodes. Together with the node numbers, a control surface nS is supplied for each edge where n is the normalized vector, normal to the surface, and S the area of the surface itself. The surface vector for each edge is oriented from the node with the first index to the node with the second index.
The sum of the surface vectors over a closed volume is equal to the null vector

$$\sum_k n_{ik} S_{ik} = 0. \tag{3.6}$$

This is a check performed by the preprocessor for each control volume of the dual grid.

To close the control volumes at boundaries, control surfaces are supplied at the boundary. In Fig. 3.2 the control surfaces to node are given at all edges connected to node 3, whereas the edges to the boundary node 1 do not close the control volume. To close the control volume $\mathcal{V}_1$, a control surface is directly set on the boundary surface. At corner points, where two or more boundaries can meet, the boundary control surface is split into sub-surfaces for each boundary condition separately. In such cases, the boundary node may occur in more than one boundary conditions.

The integral formulation of the governing equations 3.1 is numerically applied to a control volume, that surrounds the unknown at a node (0 taken as example), as

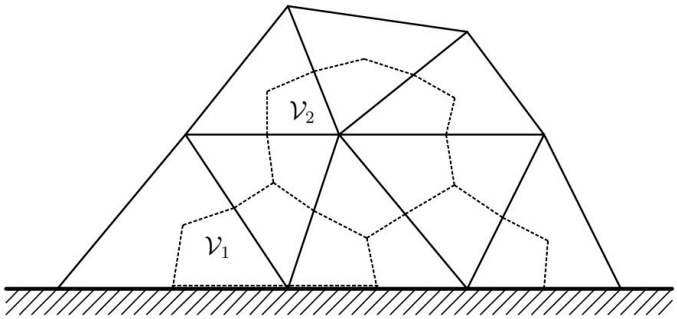


Figure 3.2 – The unstructured input grid (solid lines) and the dual grid (dashed lines) forming the control volumes at the boundary.

$$\frac{\partial}{\partial t}(q_0\mathcal{V}_0) + \sum_{k=1}^{m_0} F_{I_{0k}} n_{0k} S_{0k} - \sum_{k=1}^{m_0} F_{V_{0k}} n_{0k} S_{0k} = Q_0\mathcal{V}_0 \tag{3.7}$$

where m_0 is the number of nodes surrounding the node 0. Fluxes $F_{I_{0k}}$ and $F_{V_{0k}}$ are computed on the edge connecting node 0 and node k, being $n_{0k}S_{0k}$ given; the source terms Q_0 are directly computed at the node.

Inviscid Fluxes

The numerical scheme implemented for the inviscid fluxes F_I is a central scheme with artificial dissipation. The inviscid flux across the control surface between node 0 and node 1, is computed as

$$\boldsymbol{F}_{I_{01}} = \boldsymbol{F}_I\left(\frac{q_0 + q_1}{2}\right) - d_{01} \tag{3.8}$$

where d_{01} is the artificial dissipation. This term is chosen as a blend of second and fourth order differences, which corresponds to a blend of first and third order of differences for the fluxes. The artificial dissipation is expressed as

$$d_{01} = \left(\varepsilon_{01}^{(2)}(q_1 - q_0) - \varepsilon_{01}^{(4)}(\nabla^2 q_1 - \nabla^2 q_0)\right)\varphi_{01}\lambda_{01} \tag{3.9}$$

whith ∇^2 is the undevided Laplacian operator and where

$$\nabla^2 q_0 = \sum_{k=1}^{m_0}(q_k - q_0) = -m_0 q_0 + \sum_{k=1}^{m_0} q_k. \tag{3.10}$$

The term λ_{01} denotes the local spectral radius, defined as

$$\lambda_{01} = \left(|\boldsymbol{U}_{01}\boldsymbol{n}_{01}| + c_{01}\right) S_{01}, \tag{3.11}$$

where $\boldsymbol{U}_{01} = (\boldsymbol{U}_0 + \boldsymbol{U}_1)/2$ is the cell face speed and $c_{01} = (c_0 + c_1)/2$ is the face speed of sound.
The factor φ_{01} is introduced to account for the stretching in the grid and is defined as

$$\varphi_{01} = 4\frac{\varphi_0\varphi_1}{\varphi_0 + \varphi_1} \tag{3.12}$$

being φ_0 the ratio between the integrated spectral radius

$$\varphi_0 = \left(\frac{\lambda_0}{4\lambda_{01}}\right)^p \tag{3.13}$$

where

$$\lambda_0 = \sum_{k=1}^{m_0} \left(\boldsymbol{U}_{0k}\boldsymbol{n}_{0k} + c_{0k}\right) S_{0k} = \sum_{k=1}^{m_0} \lambda_{0k} \tag{3.14}$$

with λ_{0k} computed by 3.11. The factor $p = 0.3$ is chosen to have a close resemblance with the Martinelli eigenvalue scaling for structured grids, giving an artificial dissipation proportional to the local spectral radius λ_{0k} in the direction of the stretching. In the other directions it gives a value slightly larger than the local spectral radius [16].

The second order difference dissipation $\varepsilon_{01}^{(2)}$ is chosen to be active in the neighborhood of strong pressure gradients, like shocks, and small in other regions of the flow

$$\varepsilon_{01}^{(2)} = \kappa^{(2)} \left(\left| \sum_{k=1}^{m_0} (p_k - p_0) \right| \Big/ \sum_{k=1}^{m_0} (p_k + p_0) \right) s_2 \tag{3.15}$$

where $\kappa^{(2)}$ is a user-defined constant ans s_2 is a scaling factor to reduce the dependency on the number of sorrounding nodes.
The fourth order difference dissipation $\varepsilon_{01}^{(4)}$ is deactivated in the vicinity of shocks

$$\varepsilon_{01}^{(4)} = \max\left(0, \kappa^{(4)} - \varepsilon_{01}^{(2)}\right) s_4 \tag{3.16}$$

where $\kappa^{(4)}$ is another user-defined constant and s_4 is another scaling factor chosen in accordance to s_2 ($s_4 = s_2^2/4$). For this project's simulations, the user defined constants are set to the default values of

$$\kappa^{(2)} = 0.5, \qquad \kappa^{(4)} = 0.02. \tag{3.17}$$

Due to the computational complexity of the dissipation operator, d_{01} is effective only on the finest grid level. On coarser grids of the multigrid strategy, a simplified second order difference operator is used to save computational time and increase stability. This simplified operator is defined as

$$d_{01} = \varepsilon_{01}^{(0)}(q_1 - q_0)\varphi_{01}\lambda_{01}, \tag{3.18}$$

with

$$\varepsilon_{01}^{(0)} = \kappa^{(0)} s_2 \frac{n_{dim}}{3}, \tag{3.19}$$

where $\kappa^{(0)}$ is as well a user-defined constant, set by default equal to 0.15.

Viscous Fluxes

When resolving the viscous fluxes, in order to ease the computation, a thin-layer approximation is made. Considering the viscous stresses in the momentum equations, they can be expressed as

$$\tau_{ij} n_j = (\tau_{ij} n_j)_n + (\tau_{ij} n_j)_{tan}. \tag{3.20}$$

The first term on the right hand side contains only the normal derivatives and is the one taken into account with the thin-layer approximation, while neglecting the second term. The only considered term is formulated as

$$(\tau_{ij} n_j)_n = \mu \left(\frac{\partial U_i}{\partial n} + \frac{1}{3} \left(\frac{\partial U_j}{\partial n} n_j \right) n_i \right), \tag{3.21}$$

where the normal derivatives can be approximated on the edges as

$$\frac{\partial q_{01}}{\partial n} = \frac{q_1 - q_0}{|x_1 - x_0|}, \tag{3.22}$$

where the normal is directed from node 0 to node 1. Such formulation involves only two points in computing the normal gradients at the edges and hence automatically leads to a compact second derivative. This property can be shown recalling the identity of Laplace's equation

$$\int_\Omega \nabla^2 \phi dV = \oint_{\partial\Omega} \frac{\partial \phi}{\partial n} dS, \tag{3.23}$$

thereby the approximation of the Laplace's operator at a node (e.g. 0) reads as

$$\nabla^2 q_0 \approx \frac{1}{\mathcal{V}_0} \sum_{k=1}^{m_0} \frac{q_k - q_0}{|x_k - x_0|} S_{0k}. \tag{3.24}$$

Gradients

The full viscous terms (Eq. 3.4), as well as the turbulence production terms (Eq. 3.2), are functions of unkown variables' gradients at nodes. M-Edge offers three different ways of discretize nodal gradients. The one chosen for the presented work is by means of the Green-Gauss identity

$$\int_\Omega \nabla \phi dV = \oint_{\partial\Omega} \phi \boldsymbol{n} dS. \tag{3.25}$$

The discretization that follows is expressed as

$$\nabla q_0 \approx \frac{1}{\mathcal{V}_0} \sum_{k=1}^{m_0} \frac{1}{2}(q_k - q_0) n_{0k} S_{0k}. \tag{3.26}$$

The above approach is at least first order accurate in the inner domain on triangular and tetrahedral meshes, in two and three dimensions. On equidistant rectangular grids it is second order accurate in the inner part and first order on the boundaries [16].

3.1.3 Time Discretization

Simulations that will be presented later are all carried out as steady state time calculations. M-Edge implements an explicit time-marching Runge-Kutta scheme. Considering the general form of a time differential equation

$$\frac{dq}{dt} = R(q), \tag{3.27}$$

where $R(U)$ is the spatial differential operator, the Mutistage Runge-Kutta scheme reads

$$\begin{aligned} q^1 &= q^n + \alpha_1 \Delta t R(q^n), \\ q^2 &= q^n + \alpha_2 \Delta t R(q^1), \\ &... \\ q^p &= q^n + \Delta t R(q^{p-1}), \\ q^{n+1} &= q^p. \end{aligned} \tag{3.28}$$

The local time step is computed for each node as

$$\Delta t_0 = \min\left(\text{CFL} \frac{\mathcal{V}_0}{\lambda_0}, \text{CFLVIS} \frac{\mathcal{V}_0}{\lambda_{V_0}} \right), \tag{3.29}$$

where λ_0 is the covective spectral radius, defined as in Eq. 3.14, and λ_{V_0} is the corresponding viscous specral radius, defined as

$$\lambda_{V_0} = \sum_{k=1}^{m_0} \tilde{\mu}_{0k} 2^{n_d im} \frac{S_{0k}^2}{\rho_{0k}}, \tag{3.30}$$

with $\tilde{\mu}_{0k}$ the sum of the turbulent and dynamic viscosity. CFL and CFLVIS are respectively the inviscid and viscous *CFL* numbers. They are user inputs and if CFLVIS$<$0 it is substituted by CFL in expression 3.29. This is the case of the default setup, where CFLVIS$=$1.0.

The coefficients α_i determine the order of accuracy area and the stability of the scheme and can be chosen to better suit the problem to be solved. For steady state calculations a first order accurate scheme, which provides good smoothing for both central and upwind schemes, with 3 stages is recommended [16], with default α_i as in Table 3.1. Such scheme allows good smoothing properties, with one computation of the artificial viscosity in the first

Table 3.1 – Coefficients for 3-staged Runge-Kutta scheme.

α_1	0.66667
α_2	0.66667
α_3	1.0

stage. The default setting, in fact, is that the numerical dissipation is calculated only once, in the first stage, instead of in all Runge-Kutta stages. This way the computational cost is reduce substantially but with the drawback of need of the dissipative terms to be stored separately.

3.1.4 Multigrid Strategy

The multigrid strategy sensibly speeds up convergence. On M-Edge is possible to set up to 4 grid levels, which, as introduced previously, are created by the preprocessor. The different grid levels are referred to with the index $l = 1, ..., L$, with L the finest grid. Further than the number of levels, it is also possible to chose the multigrid method. V-, W- and F-cycle are available and are structured on 4 levels as depicted in Fig. 3.3. In the multigrid approach, the

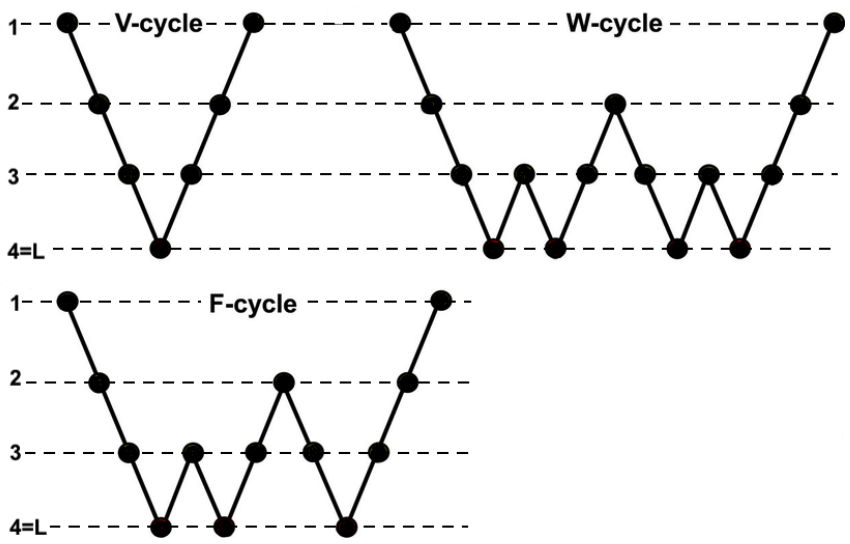


Figure 3.3 – V-, W- and F cycles of a four-level multigrid strategy.

Navier-Stokes equations are solved on the finest grid as

$$\frac{\partial q^L}{\partial t} + N_L(q^L) = 0, \tag{3.31}$$

with $N_L(q^L)$ being the spatial discretization of the Navier-Stokes operator on the finest mesh L. On the coarser grids l the problem is approximated as

$$\frac{\partial q^l}{\partial t} + N_l(q^l) = F_l, \tag{3.32}$$

with F_l the *forcing function*, defined as

$$F_l = N_l(I_{l+1}^l q^{l+1}) + \hat{I}_{l+1}^l \left[F_{l+1} - N_{l+1}(q^{l+1})\right]. \tag{3.33}$$

I_{l+1}^l and $\hat{I}_{l+1}^l$ are the *transfer operators*, of restriction type, of respectively the unknowns and the residuals from finer mesh $l+1$ to coarser mesh l. After temporal discretization, Eq. 3.31 reds

$$S\Delta q^l + N_l(q^{l(0)}) = F_l. \tag{3.34}$$

In the equation above the current solution on mesh l, obtained as $q^{l(0)} = I_{l+1}^l$, is smoothed and updated to Δq^l. The S operator is the *smoother* and corresponds to the chosen time integration method, in this case the explicit Runge-Kutta scheme. Once the solution on the coarsest mesh is smoothed, the coarse-to-fine sweep of the multigrid cycle is initiated. The current solutions on finer grids are updated with the solution on the next coarser level

$$q^l = q^l + I_{l-1}^l(q^{l-1} - I_l^{l-1} q^l), \tag{3.35}$$

with I_{l-1}^l the transfer operator, of injection type, from coarser level $l-1$ to finer level l.

In the presented work, W-cycle with 4 stages is the default choice, with one sweep of smoothing at each stage. In order to compute a better initial solution [16], a full multigrid approach is employed. This way the calculations start from coarsest grid and, after a certain condition is satisfied, it proceeds with a new two-grid cycle. In a similar way, the third and fourth grid levels are added and new cycles computed. Every cycle is controlled by both a limit on the number of iterations (500) and a convergence criteria (drop of 5 orders of magnitude in the residuals), which are both user inputs. The wall-function boundary conditions are applied only in the last cycle on the finest grid.

3.2 Pointwise

Pointwise is a mesh generator software specialized in CFD applications. It covers all aspects of preprocessing, from geometries import to meshes export directly with the proper format for selected the flow solver. High quality grids can be created quickly, intuitively and eventually through script programming. The high quality of meshes significantly improves convergence and accuracy of solvers. Pointwise can generate multiblock, structured, unstructured, overset, high-order and hybrid meshes, also using the anisotropic tetrahedral extrusion (T-Rex) technique, in-house developed for the highly automated generation of hybrid meshes for accurate resolution of boundary layers. In this section basic aspects of the software will be described, in order to give an overall knowledge of how the mesh generation works and influences the final result.

3.2.1 Geometry Model

The models on which the mesh is built are called *database*. Databases can be created both from imported CAD models and designed by scratch directly into the solver environment. In both cases, an important feature of the code is the possibility of arrange different entities, or group of entities, on different layers. Organizing the project in such way makes it more manageable and easier to access and modify single parts. Crucial for a good meshing and CFD simulation is to clean the geometry. In most cases geometry models can be less than watertight, with gaps between surfaces that are large enough to cause problems for meshing. Pointwise provides a broad range of solid modeling operations, some can be performed automatically on geometry import while others can be fine-tuned for just a portion of the model. By forming the geometry model into a watertight solid, gaps are closed implicitly and meshing can proceed with ease.

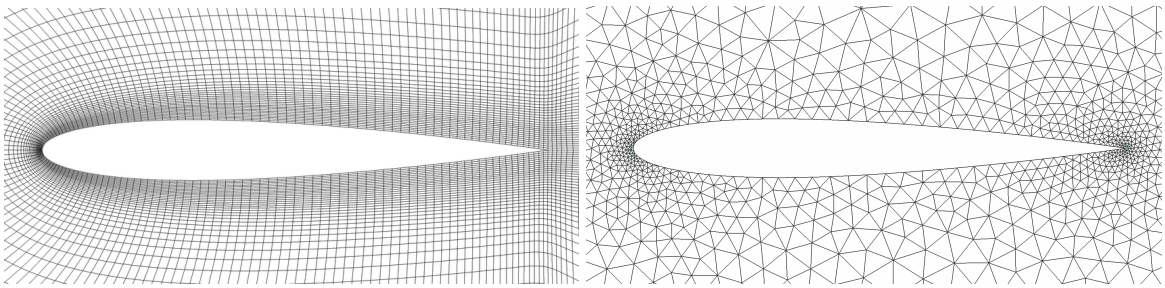
3.2.2 Meshing

Mesh generation in Pointwise is based on a hierarchical system of grid entities. These are:

- *Connectors*–1D grid entities;
- *Domains*–2D grid entities;
- *Blocks*–3D grid entities.

The strategies to build a mesh vary depending on the type of grid the generation aims for. There are basically two type of grids:

- *Structured* grids, are identified by regular connectivity among the nodes. The possible element choices are quadrilateral in 2D and hexahedra in 3D. This model is highly space efficient, since the neighbourhood relationships are defined by storage arrangement. Some other advantages of structured grid over unstructured are better convergence and higher resolution. Disadvantage is a complex grid generation, that can hardly be generally automated in a generation algorithm;



(a) Structured grid. (b) Unstructured grid.

Figure 3.4 – NACA0012 airfoil grids.

- *Unstructured* grids, are identified by irregular connectivity. It cannot easily be expressed as a two-dimensional or three-dimensional array in computer memory. This allows for any possible element that a solver might be able to use. Compared to structured meshes, the unstructured can be highly space inefficient since it calls for explicit storage of neighborhood relationships. On the other hand, with unstructured grids is way easier to cluster grid points where needed (see Fig. 3.4), without influencing the rest of the computational domain. This makes generating and refining grids on complex geometries an easier task than with structured grids. Unstructured grids typically employ triangles in 2D and tetrahedrons in 3D.

Both structured and unstructured meshes are used in this project. In the following a brief description of the steps to generate these type of grids is presented.

Structured Meshing

Before creating new connectors and domains, the user sets the type of grid and can specify some default settings. The main features that can be set in this phase are the number of grid points on the entity and the average spacing among them. The two are mutually exclusive and the choice depends on the type of mesh. The number of points is usually better suited to structured grid generation. Setting this default will have Pointwise automatically assign a fixed number of grid points for every new connector created after the default value is set or changed. Another useful default setting is the distribution of the grid points on connectors. Specifying the first spacing and a distribution function, it is possible to modify the density of grid points in a certain region. In the mesh showcased in Fig. 3.4b this feature is employed to refine the mesh around the leading and trailing edge of the airfoil. It follows a better resolution of the curvature of the solid surface and consequently of the flow deflection due to the presence of the body. When meshing grids for the presented work, tanh (hyperbolic tangent) distribution functions are used and the spacing constraint at the edge-nodes of the connectors is adapted depending on the necessity.

The hierarchical structure in Pointwise makes the workflow very straightforward: before are created the boundary entities and then they are assembled to obtain the larger entity. Mandatory in structured meshing is that two opposite-facing boundary entities have the same number of grid points. Once the larger entity is assembled, the solver algorithm is executed

to actually create the entity. All structured domains and blocks are initialized with a trans-finite interpolation (TFI), creating a first solution of the mesh; for this project the Standard TFI is used, which is arc-length based, so grid lines rarely will pass outside the boundaries of the domain or block. After initialization an elliptic PDE solver is executed to smooth the initial solution to the final mesh, in accordance to some user defined settings. Using the solver improves general orthogonality and spacing at the boundaries and cell quality in areas with significant topology skewing.

The last phase employs the *Examine* functionality to have diagnostic feedback on the solution. Such functionality provides full capability for checking grid quality - aspect ratio, skewness, orthogonality, surface proximity, etc. - through both colormaps and direct cell probes.

Unstructured Meshing

Unstructured meshing is very similar to structured meshing, hence only relevant differences are described in this part. When setting defaults for unstructured connectors, average spacing is preferable to number of points, in order to better control the distribution and density of grid points in the domain. A good practice has shown to be setting the max deviation as the 1% of the average spacing constraint at the extremes, in order to keep it close to the one imposed.

The assembly procedure presents some differences when compared to structured meshing. To assemble an unstructured domain is enough one closed connector/edge and, similarly, for an unstructured block is enough one closed domain/face. Furthermore, the *assemble special* function can be used to create unstructured domains with interior edges, just like in the airfoil case. Then, in the solver phase, the initialization clears all interior grid points and creates a new mesh by starting from the boundary points and applying the set solver attributes. These include the cell type (triangles only or triangles and quads), the solver algorithm (Delaunay triangulation is the standard setting) and restrictions on cell's size (max, min and max deviation).

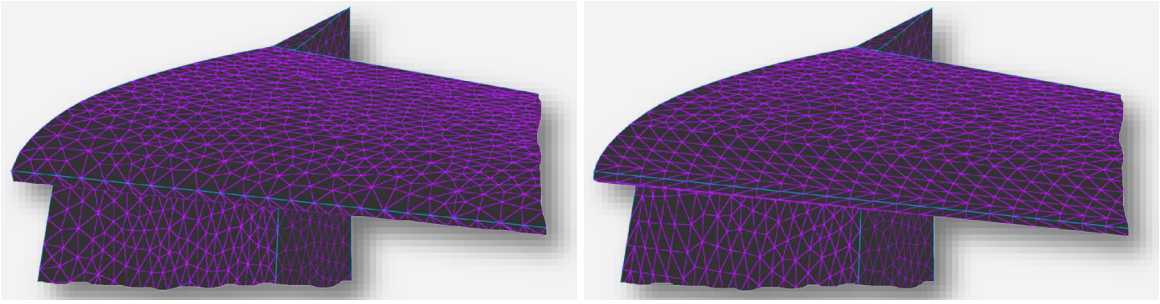


Figure 3.5 – Unstructured surface mesh without (left) and with (right) T-Rex.

Probably the most important feature of unstructured meshing is the T-Rex functionality. T-Rex stands for anisotropic tetrahedral extrusion and, as shown in Fig. 3.5, is designed to create high-quality anisotropic cells on surface domains. This Pointwise's tool has been extremely useful for this project to easily create and control near-wall cells. Previously to

the solver execution, the user can prescribe different settings that, properly tuned, can reveal pivotal for a good resolution of the near-wall gradients. Such settings are:

- *cell type* - triangles or triangles and quads in 2D, prisms or prisms and hexahedrons in 3D;
- *wall spacing* - to prescribe the right first-cell's height in order to obtain the desired y^+;
- *growth rate* - controls the growth of successive layer heights;
- *full layers* - specifies how many complete layers of anisotropic cells should generated;
- *max layers* - provides an upper limit for the desired number of anisotropic layers.

When generating a domain, T-Rex can be used to grow near-wall cells, in 2D meshing, or to better resolve a curvature, like in Fig. 3.5b, for the surface mesh of 3D projects. When creating a block of the volume mesh, the T-Rex grows anisotropic cells from the unstructured surface mesh. In this procedure, the solver algorithm makes several quality checks at each layer generation, controlling both single cells' quality and group parameters like the collision. Once the desired amount of layers is successfully generated, the isotropic tetrahedral volume mesher fills in the space between the stopped front and the outer boundary of the block with isotropic cells.

Near-wall grid size

In this section are presented and justified the choices made by the author when generating the near-wall mesh, by means of the T-Rex functionality. First, and probably most important, is the wall spacing. It is the thickness of the first cell close to the wall and its measure determines the y^+ coordinate of the first inner node. Controlling properly this variable is essential to have the desired cases to test the wall function at different ranges of y^+. Of particular interest are those ranges where standard wall functions are not consistent. This parameter is calculated by means of empirical relations for flat plate's boundary layers. Even though it can seem a rough approximation in general cases, it can be interpreted as a local assumption and has showed to be a very good practice. The first step is to calculate the Reynolds number and use it to compute the skin friction coefficient by means of the empirical relation

$$\frac{C_f}{2} \approx 0.0296 Re_\infty^{-\frac{1}{5}}. \tag{3.36}$$

The above empirical relation is derived for fully turbulent boundary layers, which is in accordance with the very high Reynolds number that characterize the cases that are tested for the presented work. Then the friction velocity is computed as

$$u_\tau = U_\infty \sqrt{\frac{C_f}{2}}, \tag{3.37}$$

and finally the wall unit is obtained by the definition of Eq. 2.14, here repeated,

$$\delta_\nu \equiv \frac{\nu}{u_\tau}. \tag{3.38}$$

The measure of the viscous lengthscale corresponds to a first cell thickness equivalent to $y^+ = 1$ and its multiples are used to set the desired y^+.

The second parameter investigated is the growth rate. It controls the growth of successive layer thickness and, indirectly, the density of cells in the near-wall region. The resolution of near-wall gradients can be sensible to this last aspect, hence different values were tested on an NACA 0012 airfoil case with the first cell height at y^+=1. Criteria of the analysis were the resolution of the skin friction coefficient, and generally of the drag coefficient, and the numerical convergence and stability. The resolution of C_f, showcased in Fig. 3.6, is very

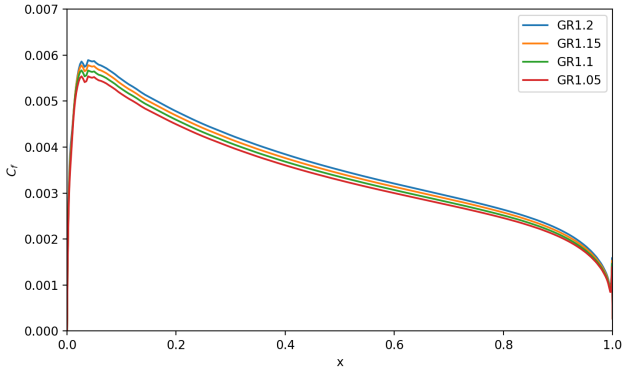
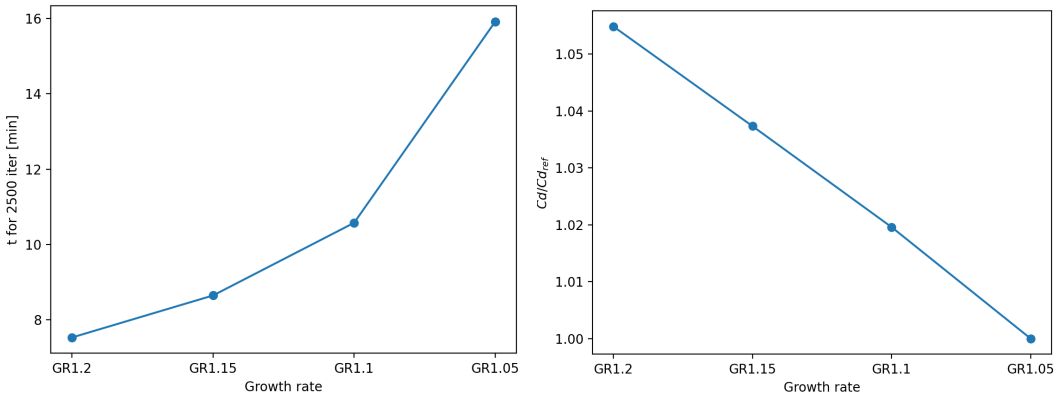


Figure 3.6 – Comparison of growth rates: skin friction coefficient along wall-tangent direction.

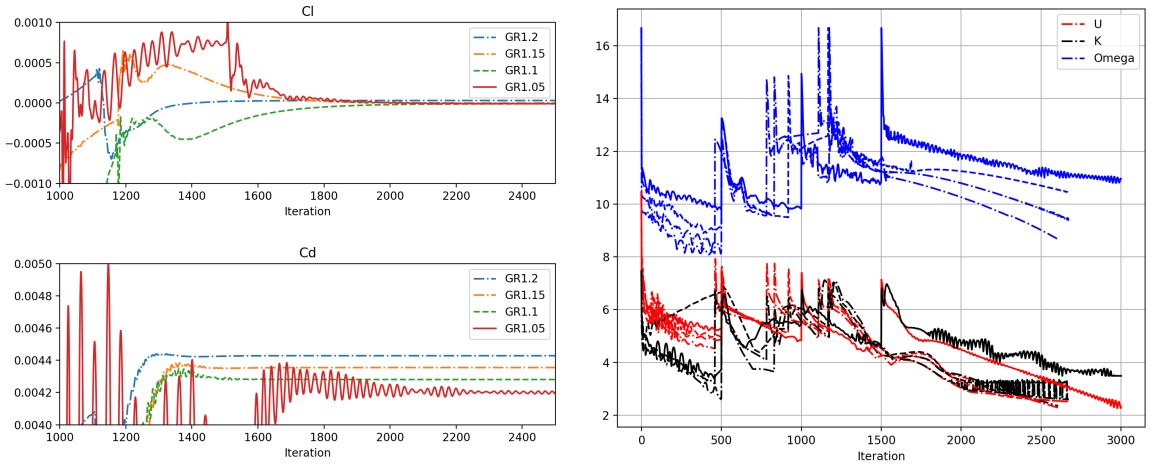
similar among the four option tested. More relevant differences are presented in the overall drag coefficient and in the computational time. Fig. 3.7a shows how the computation time required by the GR=1.2 and GR=1.15 is sensibly lower compared to the other cases. The same two values present a faster and more stable convergence. Due to the slightly more



(a) Computational time per iteration.

(b) Normalized drag coefficient.

Figure 3.7 – Comparison of growth rates: time and accuracy.



(a) Convergence of integral forces.

(b) Convergence of residuals.

Figure 3.8 – Comparison of growth rates: convergence.

accurate resolution of the drag coefficient (Fig. 3.7b), the value of 1.15 was chosen as default for the project and implemented in all the unstructured mesh generations.

The number of full layers is not as relevant as the previous two parameters but anyway a criteria to set it was formulated. It was decided to have enough layers to cover at least the whole boundary layer with anisotropic cells. To do so, again an empirical relations was employed

$$\frac{\delta}{x} \approx 0.37 Re_{\infty}^{-\frac{1}{5}}, \tag{3.39}$$

still valid for turbulent boundary layers. Introducing the characteristic length for the flow case (e.g. the chord for an airfoil), the largest boundary layer thickness is obtained, which can be used, together with the first cell thickness and the growth rate, to compute the number of layers to cover the boundary layer.

3.3 Methodology

As introduced in 1.3, the purpose of this thesis is to develop and validate an improved wall-function boundary condition that overcomes the previous formulation's weaknesses. Hence the focus of the research was not to verify the new formulation in absolute terms, maybe comparing it to experimental or other computation studies' data, but more to validate it against the previous formulation on standard cases, with the same computational settings and meshes. The justification of this choice lies in the fact that the previous formulation, and generally the implemented wall-function treatment, was extensively validated by Morgenweck in [10]. Hence, Morgenweck's work represents a consistent support for the presented project, bearing in mind that the new formulation is an improvement of the previous one and not a different and independent one. The descriptions presented in dedicated section 2.2 supports the previous statement: the improvements, both on the theoretical and implementation level, consist in simple modifications that don't change the core nature of the formulation but aim to upgrade its performances.

The accuracy level aimed to in the simulations, to compare the previous and improved formulations, was not extremely high. This because the intent of the author was to keep the numerical simulations at an easily manageable difficulty for a master's thesis. Such idea was at the base of the choices made when selecting the study-cases and, more importantly, when generating the meshes and test setups, in order to have reasonable computational time and complexity.

3.3.1 Research Process

In accordance to the thesis' aims, the research process was for the largest part of experimental nature. CFD simulations were the obvious choice of method to investigate the project's purpose, being the research's interest strictly focused on this field. Equally obvious was the choice of the simulation software, that is M-Edge, since the project concerns on how the wall functions behave and perform, implemented in this specific solver. Previously to the computational investigation, a theoretical research was performed to reach a deeper understanding of the wall-function approach and all topics that lie behind it. The so acquired knowledge has proved to be essential as a support to development of the new formulation and to better design the experimental simulations to validate it, finding the relation between the modifications and the results. When it comes to CFD simulations, many and often unknown factors influence the outcome of the computational experiment. In order to formulate reasonable conclusions on the results, it is fundamental to control, or at least be aware, of such factors. For this reason it is a diffused practice in CFD investigations to validate models on standard and extensively studied experimental cases. In the selection of validation cases, and the related meshes and setups, a reference was [17], with some adjustments to better fit the necessities of the work.

The choice made for the presented project was to approach study cases with progressively increasing complexity. This way different influencing factors can be separated and the knowledge derived from a previous case can be applied to the next. The simulations started

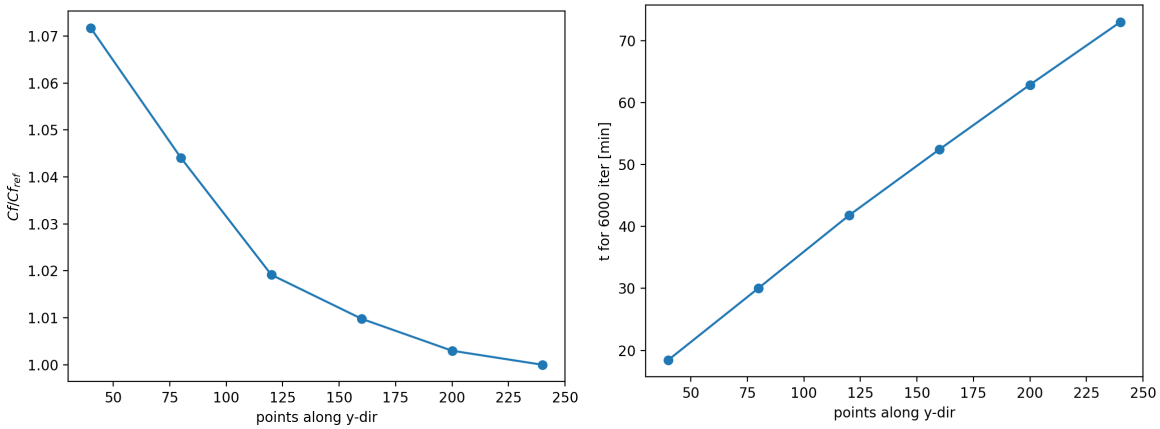
from the simplest wall-bounded flow: a zero pressure-gradient 2D flat plate, which made possible to compare the two formulations results in a very controllable case. The flat plate case made possible also to tune some computational settings comparing the results of a $y^+=1$ resolution, without wall-function boundary condition, against the empirical solution. Still to keep the case simple, the Mach number (ratio of free stream velocity to the speed of sound) was set approximately equal to 0.2, thus having an incompressible subsonic regime.
The tests continued with the flow around a 2D NACA 0012 airfoil, still in incompressible subsonic regime. At the beginning an angle of attack of 0° was selected in order to analyze the differences brought by a thick body and the consequent pressure gradients. Later increasingly angles of attack were considered in order to investigate the different stages of flow separation along the airfoil. At this point the limits of the previous formulation and the improvements brought by the new one were clearly assessed. Hence the choice was to continue to explore more complex cases with only the new formulation, to understand its performances and limits. The next step was to introduce a transonic regime with a Ma$= 0.8$. The NACA 0012 model was retained, with some adjustments to the grid. The angle of attack was set to 3°, as the interest now was in the influence of the pressure gradient brought by a shock wave and not by the separation. The last case tested was a full 3D ESAV fighter with same Mach number and angle of attack of the previous case. This last one was chosen to effectively face a case of possible industrial interest, with all the complications and difficulties related.

Every case was simulated with the three different turbulent models presented in 2.1.2, which are the Menter SST (SST), the Menter BSL + Wallin & Johansson's EARMS closure (EARSM) and the Spalart-Allmaras one-equation (SA) models. These three models were chosen as the most interesting and tuned to be used in M-Edge. The previous formulation was tested only with the SST model. For all the validation cases five different grids with different first cell-thickness were generated; these last ones corresponded to $y^+=1, 4, 10, 25, 50$. The only exception was the fighter case where, for a matter of time required, only $y^+=1, 25$ were considered.

3.3.2 Results Validation and Analysis

The first criteria to retain valid the results of a simulation was the convergence of the same. To assess that last, residuals had to be reduced of at least two orders of magnitude and reach a constant, not-growing trend. More important to monitor was the leveling out of the integral forces. If some small oscillations can be allowed for the residuals, pivotal to assess the achievement of a steady state is a complete flatness of the integral forces.

The grid can sensibly affect the outcome of a numerical simulation. Very convenient is the fact that the mesh solver in Pointwise automatically performs a lot of grid-quality checks. An important investigation to do is the grid convergence. It allows to select the right level of refinement that is a good compromise between the level of accuracy desired and a reasonable computational time. As an example, the grid convergence study performed on the flat-plate case is reported. Being implemented a structured mesh, different refinements on the wall-normal and wall-tangent directions were tested, keeping the same first cell thickness and same type of distribution. Along the y-direction (wall-normal) different amounts of grid

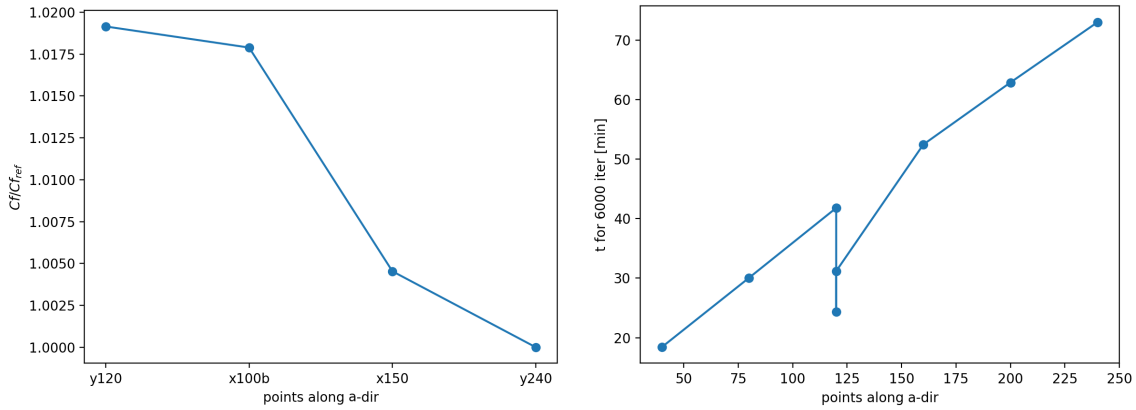


(a) Normalized C_f convergence. (b) Computational time required.

Figure 3.9 – Grid convergence (y-direction) of the skin-friction coefficient.

points were tested. The results in terms of accuracy and time are showed in Fig. 3.9. The choice fell on 120 grid points along the y-direction as the best compromise between accuracy and computational time, presenting less than 20% of error and a reduction of almost 40% of computational time, compared to the finest solution.

Similarly, different amounts of grid points were tested in the x-direction (wall-tangent). The initial 200 nodes were reduced to 150 and 100, adjusting the distribution to have a smoother growth of cells' width in the last part of the flat plate. The results obtained this way have been very useful, because was possible to improve the level of accuracy reducing the overall amount of point, hence the time required. In Fig. 3.10 is possible to appreciate how, again with reference the finest solution, the error was reduced to less than 5% and the computational time to about the 50%. In the postprocessing the results were analyzed plotting and comparing some meaningful quantities. The plots were generated by means of Python scripts, taking advantage of some libraries included in M-Edge that made possible to directly handle the FFA data format of the output files. Plots of residuals' and integral



(a) Normalized C_f convergence. (b) Computational time required.

Figure 3.10 – Grid convergence (x-direction) of the skin-friction coefficient.

forces' convergence were compared in order to assess numerical features. To examine the resolution of the near-wall flow, the particular interest when testing wall functions, the plots produced were the development of the skin friction coefficient and the pressure coefficient and wall-tangent velocity profiles at different positions along the body surface. In the plots the reference is represented by the $y^+ = 1$ solution without wall functions, against which are plotted all the different y^+ cases with wall functions. The choice of not having experimental data, or other computational studies' data, is justified by the fact that target in the postprocessing is to understand strictly the influence of wall functions on the computation. The convergence of the simulations is analyzed plotting the history of the residuals and the integral forces. The residuals are plotted with the quantity $log(\max(|R(q)|))$.

Chapter 4

Results

The general numerical setups for the tests have been described in the solver section 3.1. In this chapter every study case will be briefly introduced, with details on the specific setup. The same applies for the mesh, with the general criteria to generate the grid described in section 3.2, and only characteristic details reported here. In the introduction to each case is also described the computational domain with specifics of the boundary conditions. Inflow boundary conditions prescribe also two inflow conditions for the turbulence: the *free-stream turbulence level* (Tu) and the *viscosity ratio* (VR). The first one is related to the turbulent kinetic energy by

$$k = \frac{3}{2}(Tu\, U_\infty)^2, \tag{4.1}$$

and the second one is define as

$$VR = \frac{\nu_t}{\nu}. \tag{4.2}$$

A good estimation for the turbulence level in external aerodynamics is $Tu = 0.01$. The viscous ratio can be computed with

$$VR = \frac{L_T C_\mu \sqrt{k}}{\nu}, \tag{4.3}$$

where k can be computed with 4.1 and the turbulence length scale L_T can be estimated as the 10% of the characteristic geometrical scale of the problem.
After the brief introduction of the case, the related results are presented and discussed with the support of the postprocessing plots. These include the discussion of some coefficients that are worth to be defined here:

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho U_\infty^2}, \qquad C_l = \frac{L}{S_{ref}\frac{1}{2}\rho U_\infty^2}, \qquad C_d = \frac{D}{S_{ref}\frac{1}{2}\rho U_\infty^2} \tag{4.4}$$

which are respectively the *pressure*, *lift* and *drag coefficient*. L and D are the lift and the drag, aerodynamic forces respectively normal and parallel to the free-stream velocity direction. S_{ref} is the reference surface of the body on which the aerodynamic forces act.

4.1 Zero Pressure-Gradient Flat Plate

The turbulent flat plate case is run in essentially incompressible conditions (Ma≈0.2). The free-stream flow conditions chosen are reported in Table 4.1. The length of the flat plate is L=5 m. Such measure enables to have sensibly high Reynolds number computed with the

Table 4.1 – Flat-plate case: free-stream conditions.

Velocity	78.715	m/s
AOA	0°	
Pressure	101348.31	Pa
Temperature	300.0	K
Dynamic viscosity	$1.8531 \cdot 10^{-5}$	Pa·s
Tu	0.01	
VR	500	

free-stream conditions presented above ($Re_L = 25 \cdot 10^6$). The will to have very high Reynolds numbers is justified by the consistency of the assumptions made when calculating the first cells thickness and, more important, by the expansion of the boundary layer thickness in terms of wall units. This last feature makes flows with high Reynolds number more suitable cases to test the wall functions. The geometry of the model is depicted in Fig. 4.1, with specified the boundary conditions. Note that for this case the maximum boundary layer thickness is about

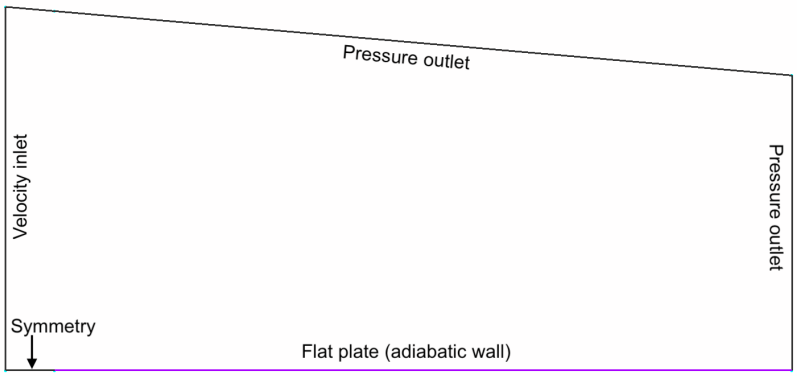


Figure 4.1 – Flat plate case: model and boundary conditions.

0.06L, so the grid height of y=L/2 is far enough away to have very little influence. The almost-rectangular shape of the computational domain is chosen to better fit a structured mesh. The upper boundary is tilted of about 5° to better suit a zero pressure-gradient condition and to avoid areas of backflow that could disturb the numerical stability. The velocity inlet condition prescribes a constant velocity profile at the boundary, hence a portion of domain is placed ahead of the plate so that there's not direct influence on the boundary layer. The wall function is set as an additional boundary condition at the adiabatic wall.

The mesh is presented in Fig. 4.2. It is a structured mesh and some details of how it was generated have been described in 3.3.2. Important is the enhanced density of grid points in

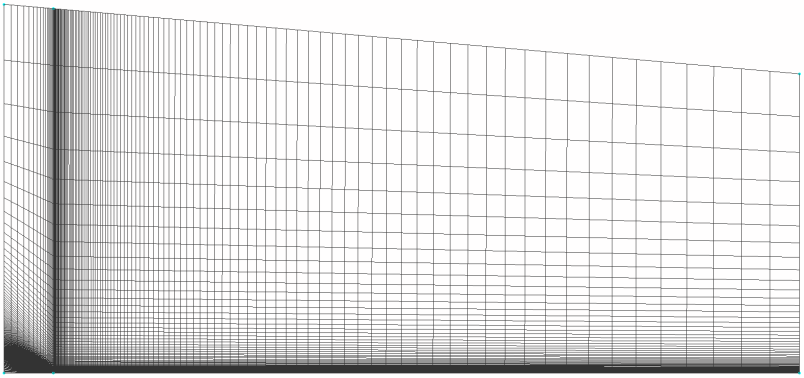


Figure 4.2 – Flat plate case: Mesh, 170 x 120 points.

the proximity of the leading edge of the flat plate, in order to have a smoother resolution of the boundary layer development. Another relevant adjustment is to avoid small thicknesses at the bottom of the inflow part. Having the same small first cells thickness as on the flat plate can result in a too stiff condition that would slow down the convergence of the simulation.

4.1.1 Preconditioning Study

In order to overcome problems with stiffness of Navier-Stokes equations in low speed flows M-Edge suggests to setup a low speed preconditioning. Having a low Mach number, such option was tested before moving on to the full simulations of the case. With preconditioning the integral formulation of the governing equations is modified as

$$M P_{\bar{q}}^{-1} M^{-1} \int_{\Omega} \frac{\partial q}{\partial t} dV + \oint_{\partial\Omega} \boldsymbol{F}_I \boldsymbol{n} dS = \oint_{\partial\Omega} \boldsymbol{F}_V \boldsymbol{n} dS + \int_{\Omega} Q dV, \tag{4.5}$$

where $P_{\bar{q}}$ is the preconditioning matrix for the pressure temperature variables $\bar{q} = (p, u, v, w, T)$. $M = \partial q / \partial \bar{q}$ represents the transformation matrix between conservative and pressure temperature variables. More details on the preconditioning matrix can be found in [16]. The preconditioning option, tested on the finest grid with $y^+ = 1$ and without wall functions,

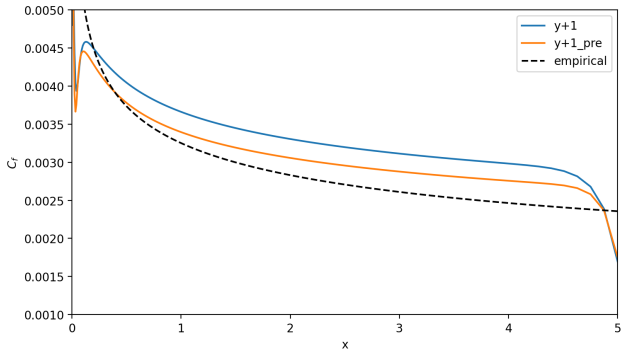
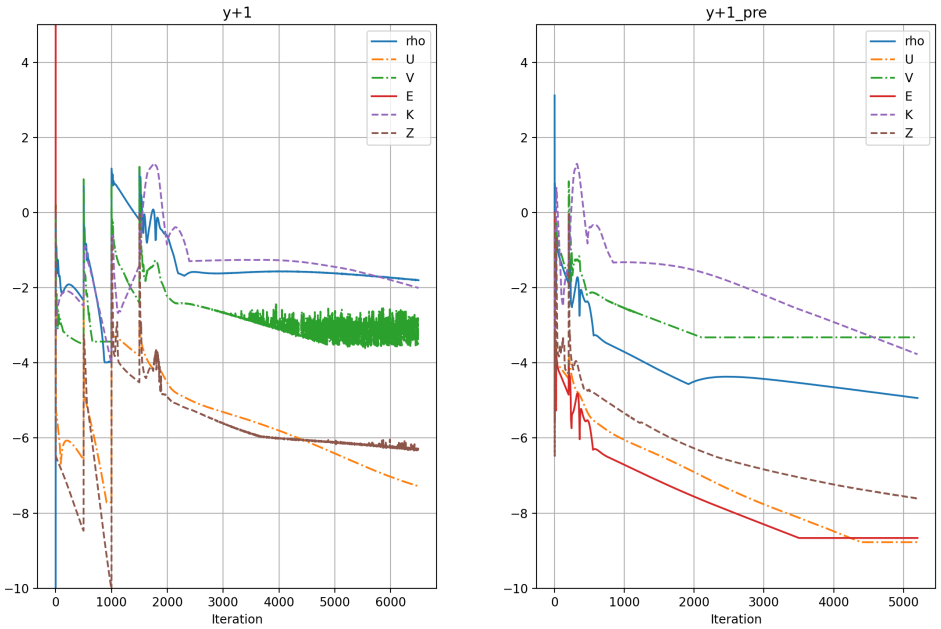


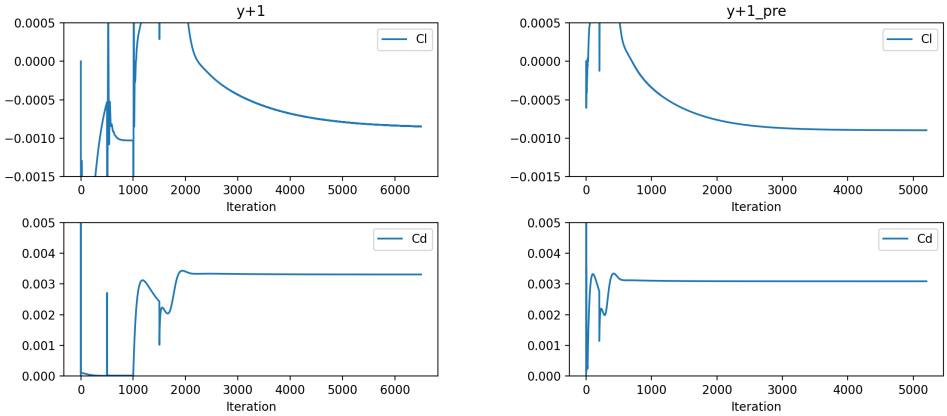
Figure 4.3 – Preconditioning test: Skin friction coefficient along plat plate surface.

shows a relevant gain both in accuracy and convergence. The yellow line in Fig. 4.3 represents the simulation with the preconditioning turned on and presents a resolution of C_f closer to the empirical law. Definitely more relevant is the improvement in convergence. Considering the plots in Fig. 4.4, with the preconditioning option the bottom level of residuals is on average reduced of two orders of magnitude. Furthermore the convergence of the integral quantities is much faster, especially the one of the lift coefficient, which is reduced by almost 3000 iterations.

In light of the above results, the preconditioning is implemented for all the cases with a low Mach number.



(a) Residuals.

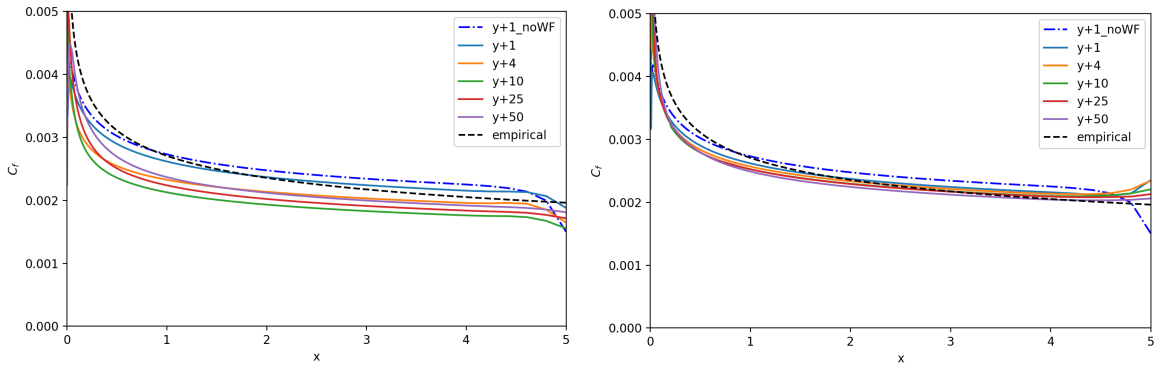


(b) Integral forces.

Figure 4.4 – Preconditioning test: convergence analysis.

4.1.2 Previous and Improved Formulation

Now the results of the previous and improved formulation, tested with the SST turbulence model, are presented. The new formulation immediately presents evident improvements in the resolution of the skin friction coefficient, displayed in Fig 4.5. With the old wall function,

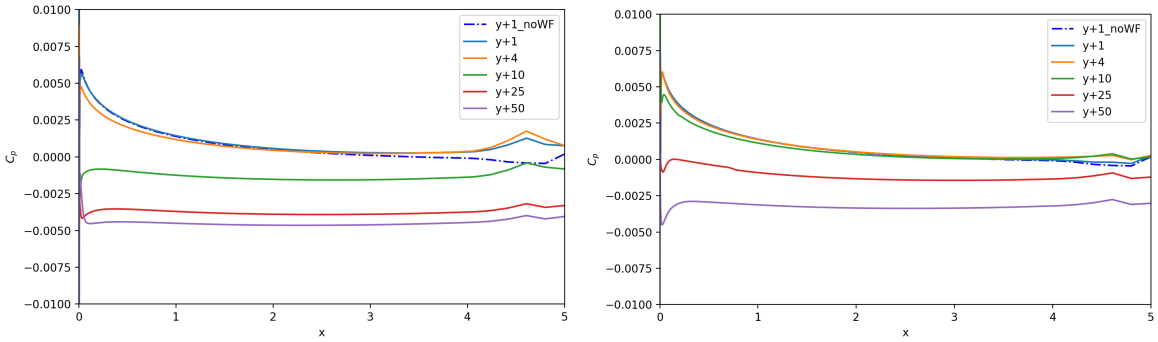


(a) Previous formulation.

(b) New formulation.

Figure 4.5 – Formulations comparison: skin friction coefficient along the flat plate surface.

the more the y^+ value at wall is large, the more the C_f prediction is far from the finest solution, reaching much smaller values. The situation is significantly improved with the new formulation, having all the different cases much closer among themselves and to the reference solution without wall function. Such improvement can be a consequence both of a better prediction of u_τ by the new formulation and of the fact that the wall stress is directly set as a flux. This last feature is reflected also in the resolution of the pressure coefficient at wall. In the plots below is possible to appreciate how, with the improved formulation, also the



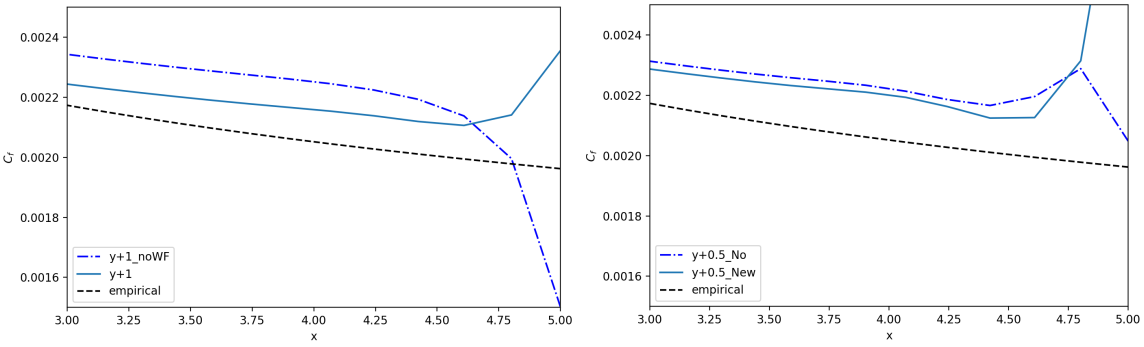
(a) Previous formulation.

(b) New formulation.

Figure 4.6 – Formulations comparison: pressure coefficient along the flat plate.

$y^+ = 10$ solution fits the finer ones and the error for the two coarser cases is reduced. Even if reduced, these errors are still relevantly large and the cause probably is a bad resolution of the first stages of the boundary layer at the leading edge. The adaption region between the inflow boundary and the flat plate is perhaps too small for the coarser cases and has a bed contribution to the overall solution.

An aspect worth to explore more is the gap between the C_f curve of the $y^+ = 1$ case with no and new wall function. One would expect a better match of the two curves, especially looking at the good match among the different grid cases. Further investigation was done, computing the same flow case but on a much finer grid, with the double of grid points along the y-direction. The results of this test (Fig. 4.7) show that the two predictions converge to a



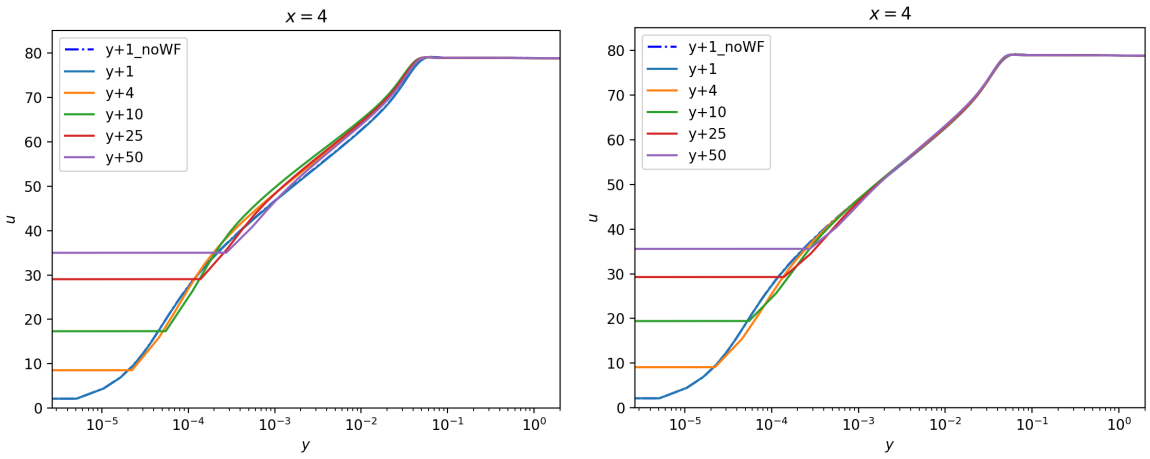
(a) Reference grid.

(b) Refined grid.

Figure 4.7 – Flat plate case: resolution of the skin friction coefficient with no and new wall function.

matching with the refined mesh. We can hence conclude that the gap is only a matter of mesh refinement and not of the consistency of the model. Moreover, considering the small measure of the gap, both in percentage and in absolute terms, we can guess that this grid-refinement issue is of relevant importance only when small measures of skin friction are resolved.

Improvements by the new formulation are presented also in the wall-tangent velocity profiles. The profiles presented in figure below are extrapolated on the line at $x = 4$ (80% of the flat plate) where the flow presents a local Reynolds number of 20 millions and the boundary layer is largely developed. The logarithmic scale on the y-axis is chosen to

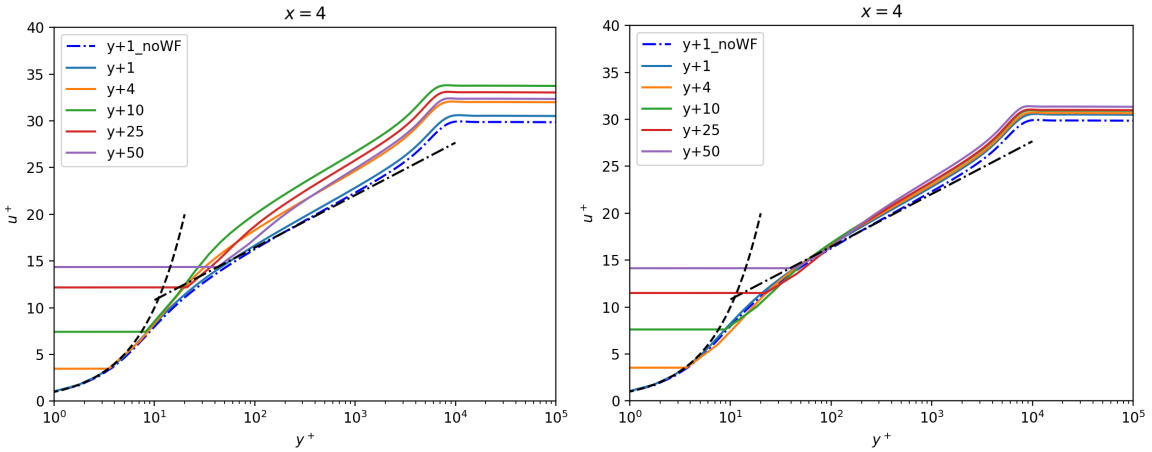


(a) Previous formulation.

(b) New formulation.

Figure 4.8 – Formulations comparison: unscaled U-velocity profiles, semilogarithmic axes.

give more relevance to the resolution in the near-wall region. The better overlapping of the different curves on the higher part of the profile proves the better quality in the modelling of near-wall velocity by the improved formulation. Improved is also the position of the bottom kinks in the curves of the cases with $y^+>1$. Those kinks represent the velocity computed at the first inner node and the ideal condition is to be located on the finest solution's profile. The performances in resolving the skin friction coefficient and the near-wall velocity come together in the velocity profiles scaled with the wall units. Such type of plot is very useful when validating wall functions as it directly shows the consistency of the solution with the modelling based on the law of the wall. In accordance to that, plots in Fig. 4.9 show clearly the improvement in accuracy brought by the new formulation. The black dashed lines in the plots represent the linear relation and log law introduced in section 2.1.1. What one seeks in these plots is a good fit of the scaled profiles with the two theoretical relations in the regions of validity. This type of accordance is also a good check for the scaling, as it is done manually in the postprocessing.

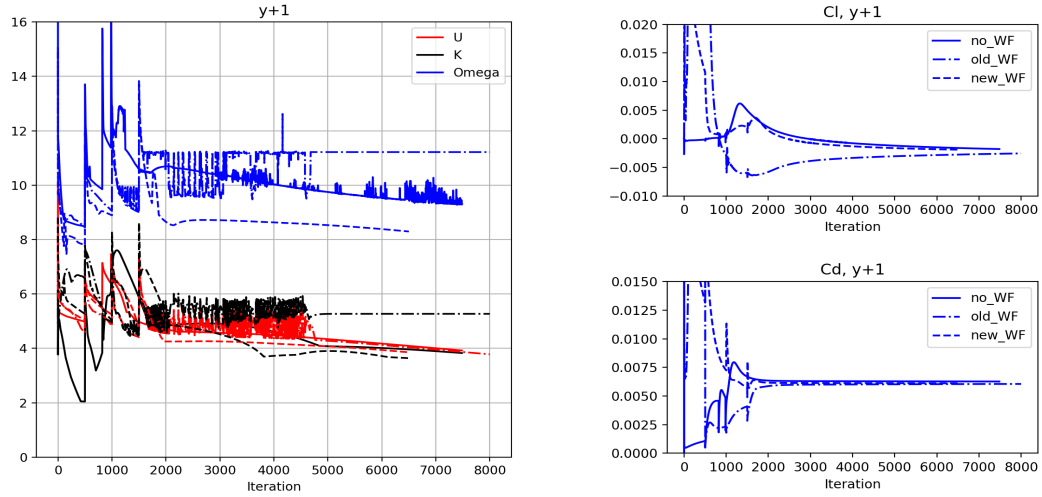


(a) Previous formulation.

(b) New formulation.

Figure 4.9 – Formulations comparison: scaled U-velocity profiles, semilogarithmic axes. Black dashed lines: linear and log law

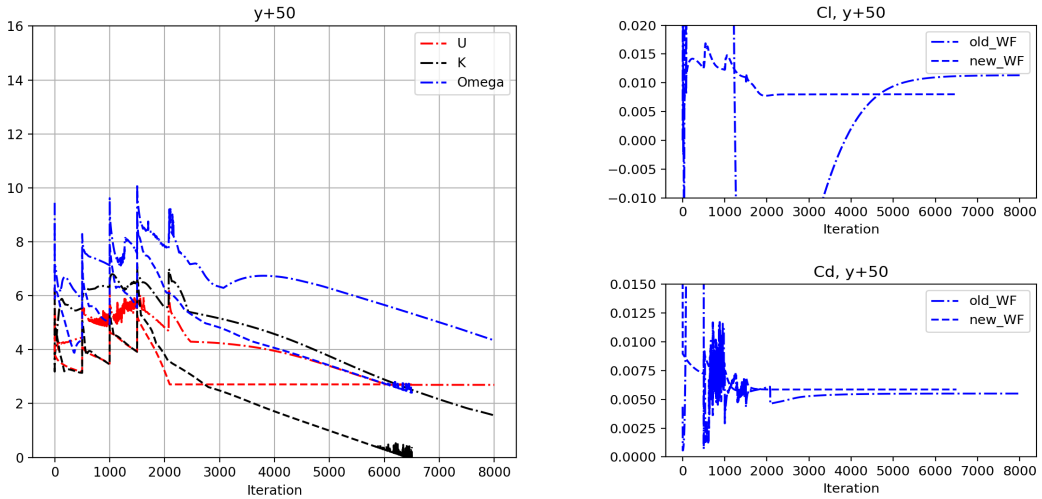
Relevant considerations can be made looking at the convergence of the simulations. The rate of convergence of the case with $y^+=1$ case does not change a lot with the wall-function boundary conditions. What changes is the value of convergence of the residuals: with the improved wall function the residuals of the turbulent quantities k and ω present a reduction of one order of magnitude, compared to without wall function, and a reduction of two orders of magnitude, compared to with previous formulation. Even if using wall functions with a mesh that has first cell thickness of $y^+=1$ doesn't seem an useful choice, checking how a wall function behaves also with such refined grids is a good way to better assess that grid adaptability which is a great weakness of standard wall-function approaches. In the test case here discussed, looking at Fig. 4.10, it can be said that the improved wall function approach doesn't affect sensibly the overall accuracy, as the asymptotic value of the integral forces is the same, with the lift coefficient converging to a zero value as expected for this flow case.



(a) Residuals. (b) Integral forces.

Figure 4.10 – Formulations comparison: convergence of y^+=1 case with no (solid lines), old (dash-dotted lines) and new (dashed lines) wall function.

More relevant differences are present in the convergence of the y^+=50 cases. Even though none of the two is accurate in the prediction of the lift coefficient, the implementation of the new wall function leads to a much faster convergence (around 4000 iteration faster) that can be observed in Fig. 4.11b. The lift resolution is strictly related to the resolution of the pressure field close to the body surface. This is probably positively influenced by the way the skin friction is prescribed as a flux in the boundary condition, giving more stability in the balance of Eq. 2.48, that includes the pressure. Fig. 4.11a confirms once again how the magnitude of the residuals is sensibly reduced with the improved formulation.

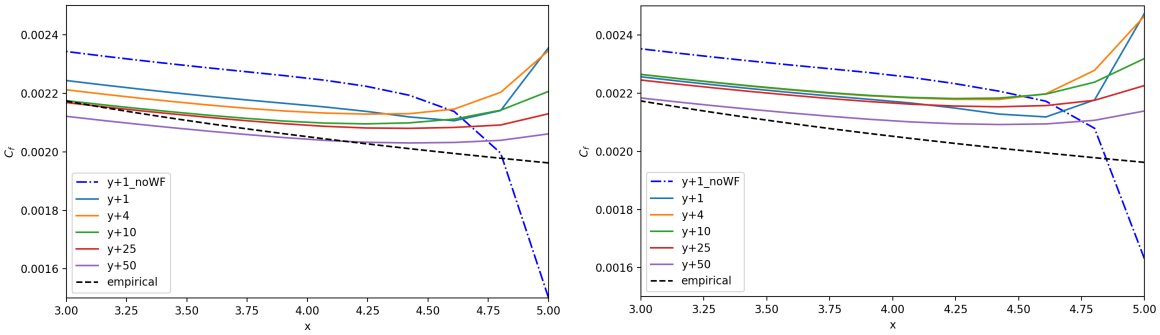


(a) Residuals. (b) Integral forces.

Figure 4.11 – Formulations comparison: convergence of y^+=50 case with old (dash-dotted lines) and new (dashed lines) wall function.

4.1.3 Other Turbulence Models

Results of the other two turbulence models employed are presented and analyzed with reference the solution with the improved formulation and the SST model.
The EARSM model formulation, described in section 2.1.2, is very similar to the SST model. Thus is not surprising that the two give very similar solutions in the most cases. This is what happens in the presented flat plate case, with no substantial differences in the overall solution. The only appreciable difference is a slightly better accuracy when resolving the skin friction

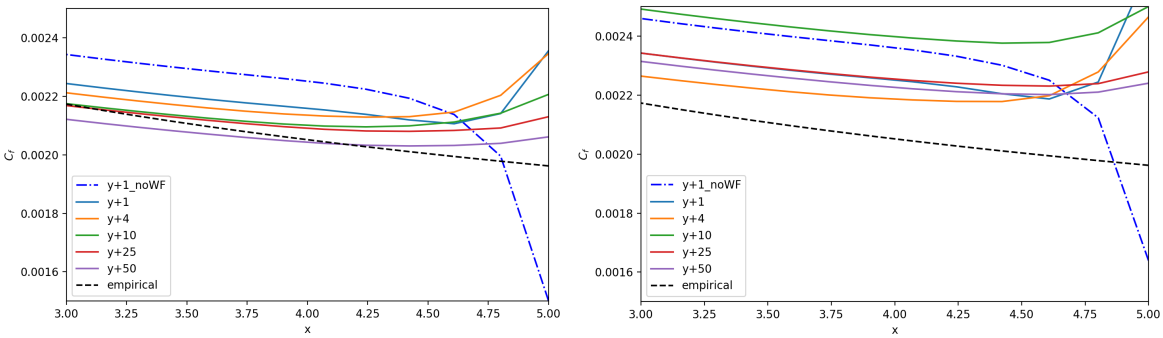


(a) SST model. (b) EARSM model.

Figure 4.12 – Improved formulation: skin friction coefficient in the proximity of x=4.

coefficient. With the EARSM model the C_f curves of the cases with y^+>1 are closer to the one with y^+=1, with only the one with y^+=50 presenting an observable gap.

Different is the situation with the SA model. Here the different nature of the turbulence model, and the consequent differences in the wall-function boundary conditions, reflect into differences in the solution. The first to notice is a different trend in accuracy of the skin



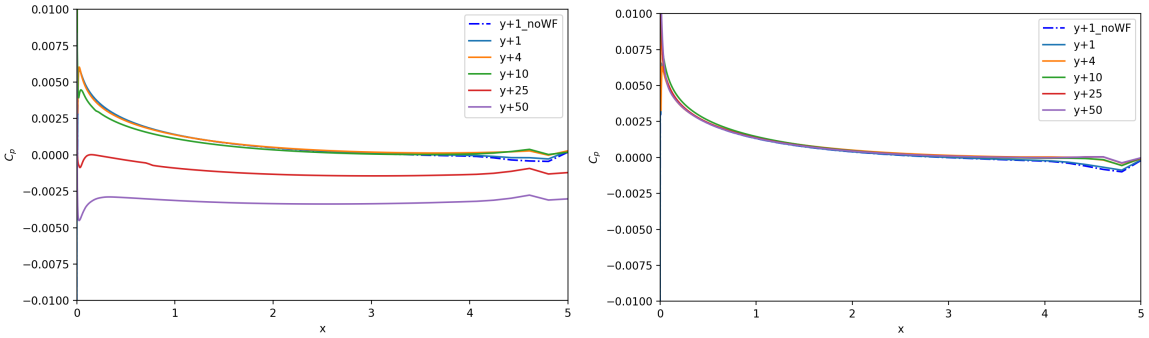
(a) SST model. (b) SA model.

Figure 4.13 – Improved formulation: skin friction coefficient in the proximity of x=4.

friction coefficient. With the SA model, the cases with y^+=1, 10 and 50 present very similar resolutions, while with y^+=4 and 25 present relevant differences. Furthermore in Fig. 4.13b we can observe how the predicted values of C_f with the SA model are a bit larger and more far from the empirical curve than with the SST model. These values are anyway much better

than the one predicted with the previous formulation, thus the performances of the new wall function for Spalart-Allmaras models are really valuable.

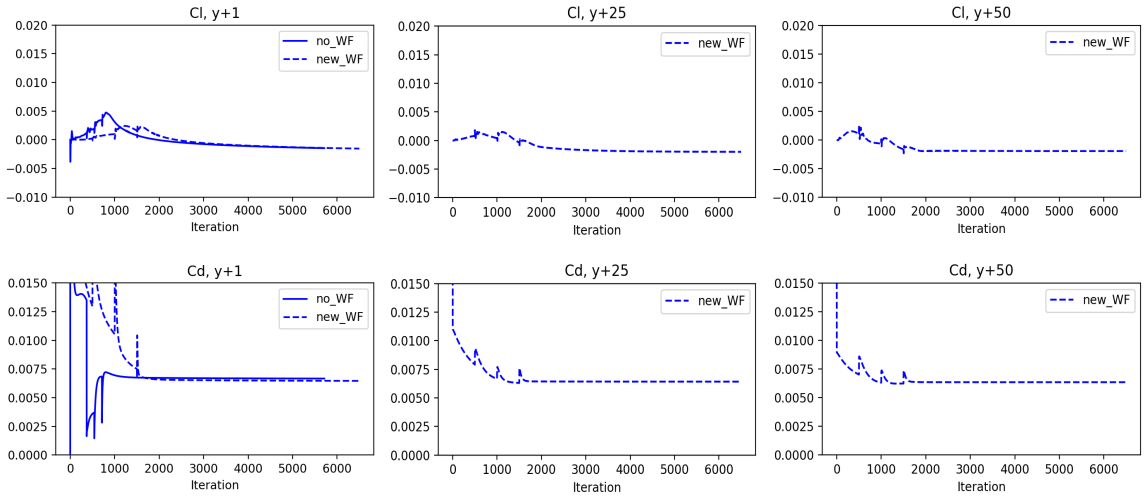
Even more valuable is the resolution of the pressure coefficient at wall. In Fig. 4.14 it is possible to appreciate how the inconsistencies of the $k-\omega$ models in resolving the pressure field for the cases with high y^+ disappear. This results in a way better resolution for the



(a) SST model. (b) SA model.

Figure 4.14 – Improved formulation: pressure coefficient along the flat plate.

lift coefficient, with now all the different cases predicting the same value as the reference solution with y^+=1 and no wall functions. Fig. 4.15 shows the convergence of lift and drag coefficients and how theses reach the same constant value. The quality of the convergence with the SA model is very similar to the previously discussed models, with the integral forces leveling out around 2000 iterations.



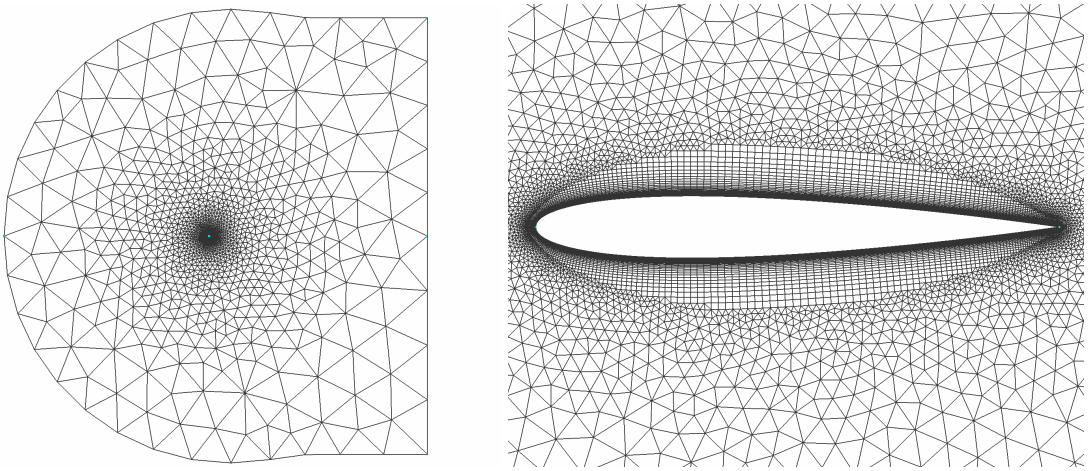
(a) y^+=1. (b) y^+=25. (c) y^+=50.

Figure 4.15 – Improved formulation, SA model: convergence of the integral forces.

4.2 NACA 0012 Airfoil, Ma=0.2, AOA=0°

The NACA 0012 is a symmetric airfoil with its maximum thickness at the 12% of the chord. The free-stream conditions for this study case are the same as the flat plate case, reported in Table 4.1, keeping the incompressible regime with Ma=0.2. Likewise the flat plate, the airfoil is designed with a chord c=5 m in order to have an overall Reynolds number of 25 millions. In this section the case with 0° angle of attack is tested while in the next one higher angle of attacks are considered to approach separation phenomena.

In Fig. 4.16 is presented the computational domain for this study case. The boundaries are designed for a far-field condition. In M-Edge it is implemented as a weak characteristic boundary condition and it is mainly intended for external aerodynamics. It is based on a one-dimensional analysis of the Euler equations in the direction normal to the boundary. Depending on the sign of the eigenvalues, the characteristics are either set from free-



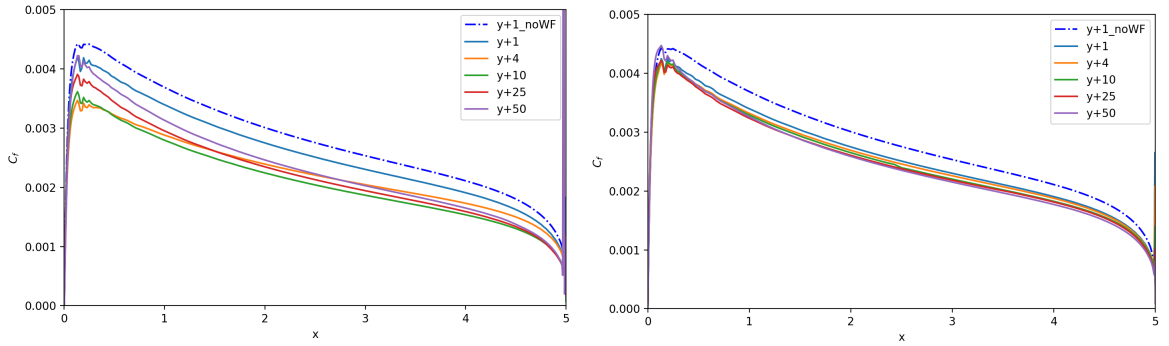
(a) Far-field region. (b) Near-body region.

Figure 4.16 – NACA 0012 case: Mesh.

stream quantities for ingoing characteristics or extrapolated from the interior for outgoing characteristics. At the airfoil surface an adiabatic wall boundary condition is retained, with optional wall functions. To minimize issues associated with effect of the far-field boundary (which can particularly influence drag and lift levels at high-lift conditions), the far-field boundary has been located almost 500 chords away from the airfoil. Even if it is a big measure, the impact on the mesh expense is moderate as in the far-field region the grid is very coarse (the spacing of grid points at the boundary is almost 100 chords). The mesh is of unstructured type, with the the cell size growing smoothly from the near-body to the far-field region. The grid is very refined around the airfoil, with 300 grid points on the body-surface connectors. These points are more densely distributed around the leading and the sharp trailing edge in order to better resolve the surface curvature and the regions in proximity of the edges. In Fig. 4.16b is observable the growth of anisotropic layers from the wall generated with the T-Rex functionality, with the procedure described in section 3.2.2. The mesh displayed in figure is the finest one with y^+=1 and counts 23005 grid points.

4.2.1 Formulations Comparison

The comparison between the previous and the improved formulation, with SST model, produce in overall the same results as in the flat plate case. The accuracy of the skin friction coefficient is sensibly improved for the cases with $y^+>1$. An interesting observation that can be made with Fig. 4.17 concerns the peaks at the trailing edge. Such numerical instabilities



(a) Previous formulation.

(b) Improved formulation.

Figure 4.17 – Formulations comparison: skin friction coefficient on the airfoil surface.

are caused by the numerical discontinuity that a sharp trailing edge is. Remarkable is how such peaks are way more low with the improved formulation. This result is an evidence of how the new wall function is numerically more stable than the previous one. The poor accuracy of C_p observed in the flat plate results, when adopting high y^+ grids, does not occur for the wing profile. Fig. 4.18 shows a full agreement among the different cases for the new formulation. Thus the hypothesis that, in the flat plate case, the poor accuracy was caused by the small distance of the boundaries is somehow confirmed.

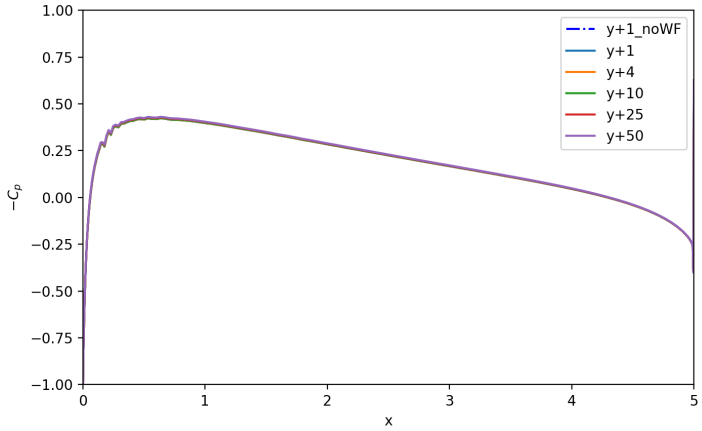
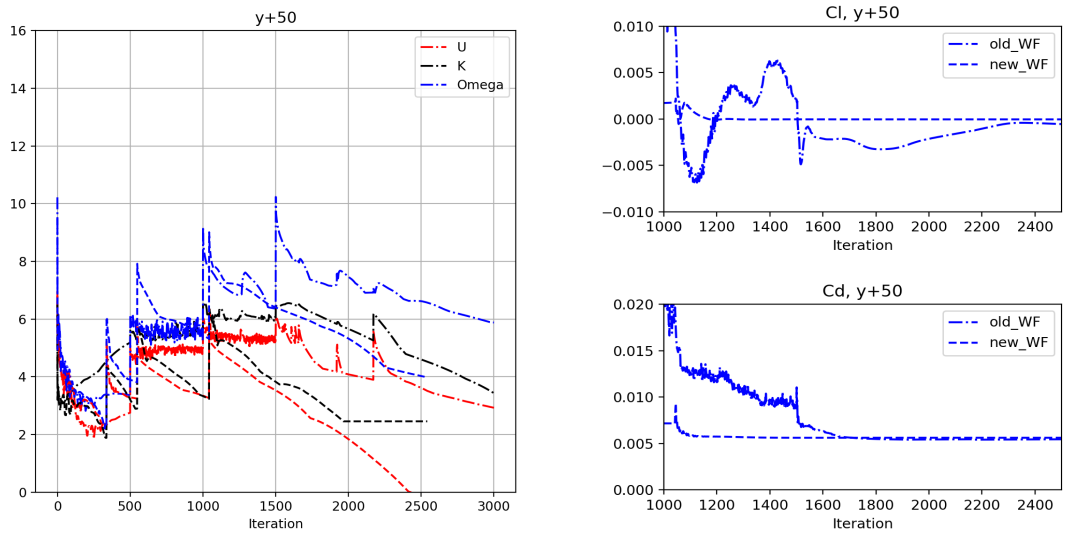


Figure 4.18 – Improved formulation: pressure coefficient on the airfoil surface

Also the improvements in terms of convergence are reaffirmed by the airfoil case's results. Looking at the convergence plots for the y^+=50 case (Fig. 4.19), with the improved formulation the residuals reach lower order of magnitudes and the iterations required to



(a) Residuals.

(b) Integral forces.

Figure 4.19 – Formulations comparison: convergence of y^+=50 case with old (dash-dotted lines) and new (dashed lines) wall function.

converge are sensibly reduced. Besides a proof of a better numerical stability, the reduction in the number of iterations needed represents a gain in computational time. The gain becomes more significant if we look at the time per iteration needed by each case.

Table 4.2 – NACA 0012 case, SST model: number of grid points and computational time (minutes per 1000 iterations) used by the different y^+ cases, when implementing the different formulations.

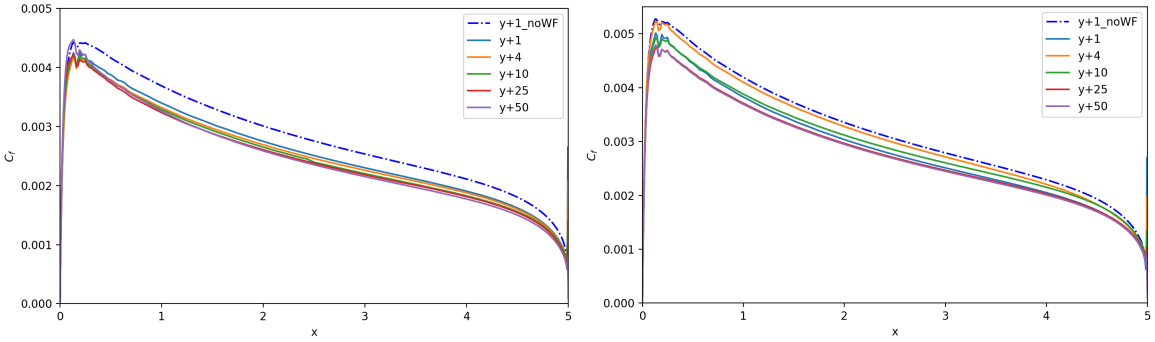
y^+	1	4	10	25	50
Number of grid points	23005	20040	18041	16121	14636
No wall function	4.60	/	/	/	/
Old wall function	4.03	3.48	3.09	2.71	2.39
New wall function	3.96	3.74	3.38	3.10	2.94

Very significant is the reduction of time brought by the improved formulation. If compared to the simulation with no wall functions and y^+=1, the computational time is halved retaining the same level of accuracy in predicting the integral forces. Some considerations should be made also for the y^+=50 case: this case with the improved wall-function boundary condition allows to obtain a very accurate solution in terms of lift and drag coefficients, as in Fig. 4.19b C_l is zero as expected for a symmetrical airfoil with zero angle of attack. The accurate resolution is obtained with around the 64% of the time expense required with the finest grid and no wall function, but increases to around the 123% if compared to the same grid with the old wall function. This last comparison is adjusted to almost a 62% if considering the fact the new formulation converges with the half of the iterations. Such computational-time reduction thanks to the improved formulation is very remarkable.

4.2.2 Other Turbulence Models

For the EARSM model, the results comparison is in fully accordance with what discussed in the flat plate case, both in accuracy and convergence terms. Again, the only appreciable difference between the results of the EARSM and SST models is a slightly better resolution of the skin friction coefficient.

Like the previous study case, more differences exist between the results of the SST and SA models. Similarly to what observed in the flat plate case, the SA models predicts higher values for the skin friction coefficient end the consistency among the different y^+ cases is a bit more poor. Looking at Fig. 4.20 is possible to affirm that the overprediction of C_f is something related more to the turbulence model than the wall-function boundary condition. Proof of this are the blue dash-dotted lines that represent the case without usage of wall functions. As in this particular situation the total drag has a bigger contribution from the skin friction than from the pressure field, the overestimation of C_f results in a C_d 15% higher for the SA model. The scatter between the different curves of C_f with the SA model is reflected in the



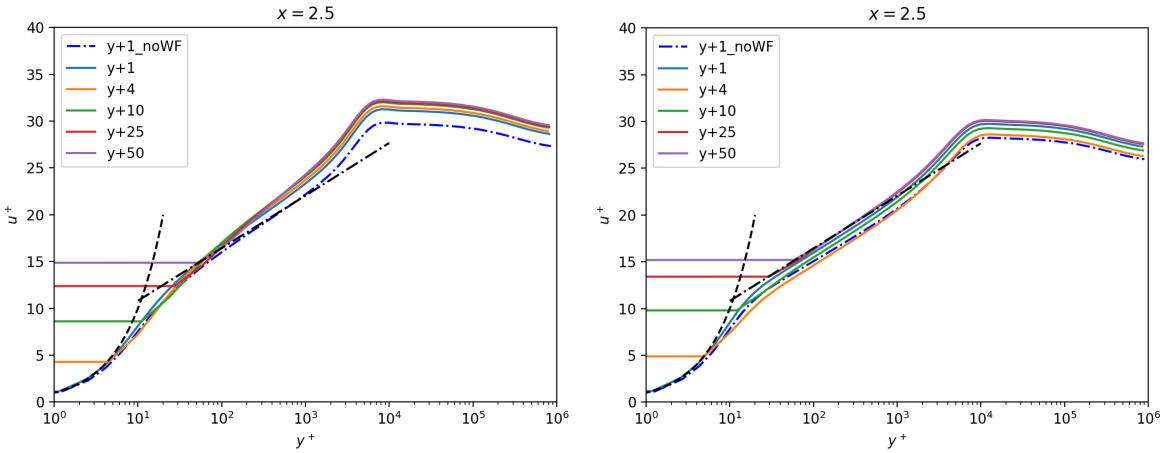
(a) SST model.

(b) SA model.

Figure 4.20 – Improved formulation: skin friction coefficient on the airfoil surface.

scaled velocity profiles in Fig. 4.21b. In the same group of plots, one difference to notice is the slope of the scaled profiles in the log region. The profiles with the SA model better fit the slope of the theoretical log law, compared to the SST model. The motivations of this aspect can be found in the different turbulence modelling as also the "noWF" cases present a different slope. There may be also some influence of the wall-function boundary conditions as the mismatch in the slope seems to accentuate with the wall functions. In fact the two wall functions are formulated differently as are intended for different turbulence models. A consequence of this is a different position of the kinks in the profile, reasonable as the two wall functions seek for slightly different theoretical profiles, as discussed in section 2.2.2.

The convergence of the SA cases is very similar to what seen with the other turbulence models confirming the improvement in the numerical stability, compared to the previous formulation.



(a) SST model, scaled profiles.

(b) SA model, scaled profiles.

Figure 4.21 – Improved formulation: wall-tangent velocity profiles at the 50% of the chord, semilogarithmic axes. Black dashed lines: linear and log law.

4.3 NACA 0012 Airfoil, Ma=0.2, high angles of attack

After 0° progressively increasing angles of attack have been simulated on the finest grid, with the SST model and without wall functions, to reach and analyze a condition of separated flow. The study on the different angles of attack, in the range from 6° to 19°, is reported in the Appendix A. The possible separation was monitored checking the skin friction coefficient, looking for negative values. In this sections only the results of the cases that showed some relevance for the purpose of the section will be discussed, which are the angles of attack of 15° and 18°. For both cases the meshes utilized and the free-stream conditions are the same as in the 0° angle of attack case. The angle of attack is set up in the solver adjusting the x- and y-component of the free-stream velocity in such a way that the resulting vector forms with the chord an angle equal to the desired angle of attack.

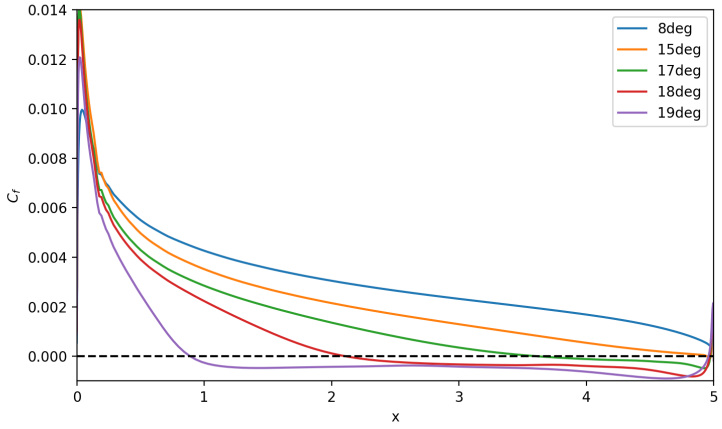


Figure 4.22 – High angles of attack: skin friction coefficient along the airfoil.

4.3.1 Angle of attack = 15°

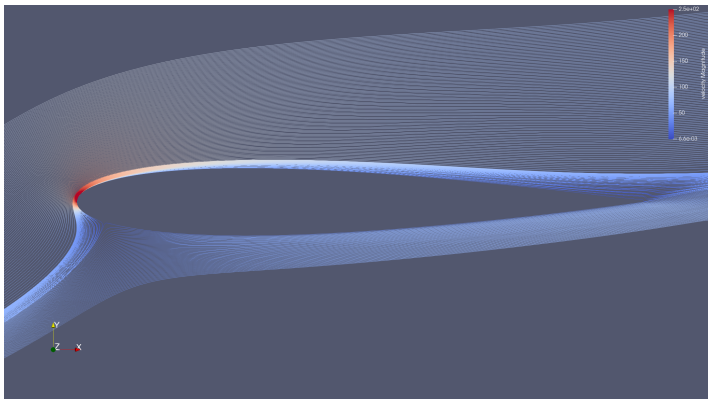
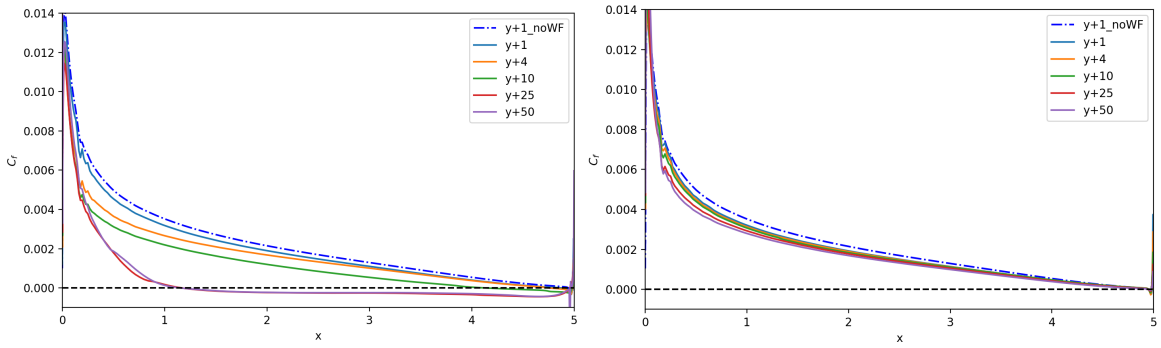


Figure 4.23 – AOA = 15°: streamlines colored by velocity magnitude.

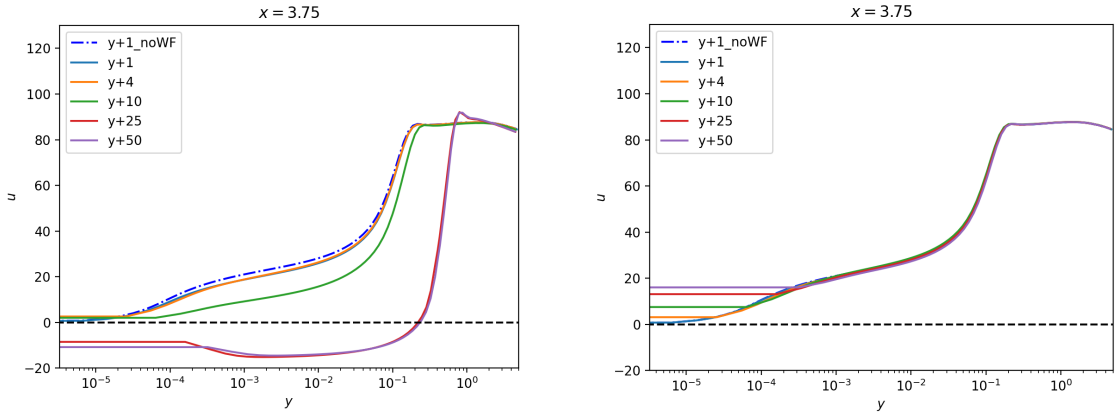
The flow solution for the present case is showed in Fig. 4.23. The flow does not really present a separation but the it is starting to detach very close to the trailing edge. Such condition is indicated also by the plot in Fig. 4.22, where the curve of 15° starts to assume negative values of skin friction coefficient very close to the airfoil's trailing edge. The threshold situation present in the flow with this particular angle of attack demonstrated to be interesting to show the previous formulation's weakness. The poor accuracy of the previous



(a) Previous formulation. (b) Improved formulation.

Figure 4.24 – Formulations comparison: skin friction coefficient on the airfoil surface with AOA=15°.

formulation when resolving the skin friction coefficient causes a wrong resolution of the flow, predicting a detached flow where there's not. Fig. 4.24 showcases this behaviour, with an early separation for the cases with y^+=25 and 50, which is obviously inconsistent with the solution on the finer grids. The improved formulation corrects this behavior, as it predicts the skin friction coefficient with good accuracy and agreement with the reference solution, like observed in the previous study cases. To notice is also that the gap previously presented between the y^+=1 solutions with no and new wall function is sensibly reduces as the resolved

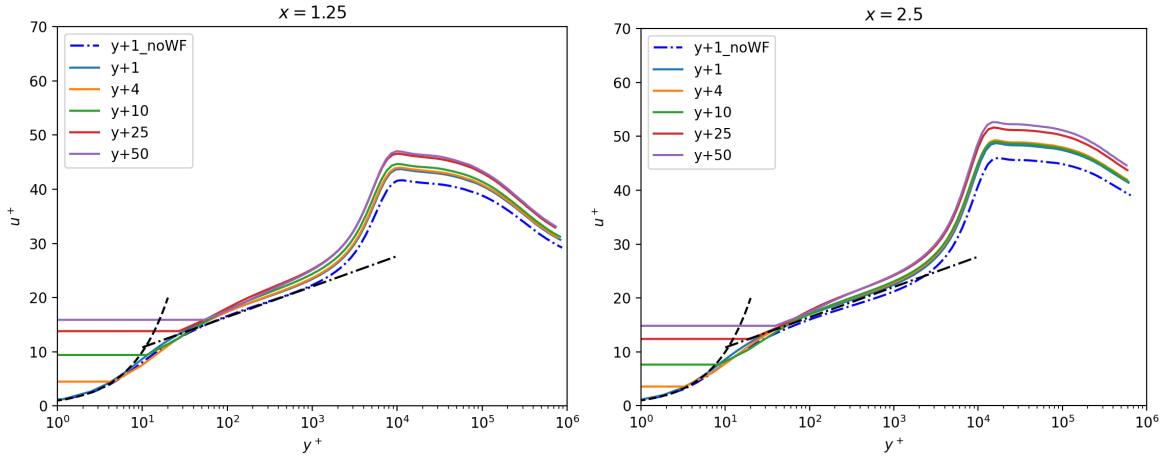


(a) Previous formulation. (b) Improved formulation.

Figure 4.25 – Formulations comparison: unscaled wall-tangent velocity profiles, semilogarithmic axes.

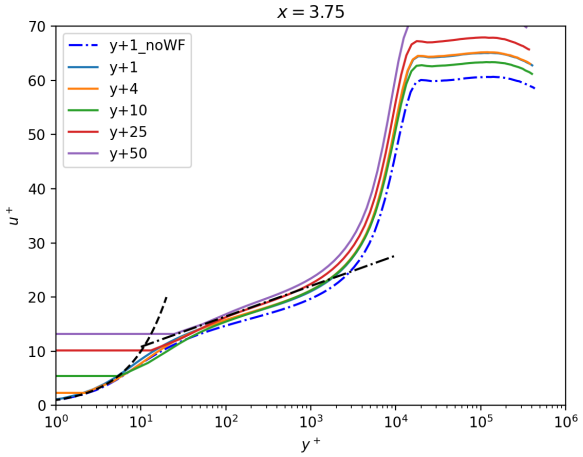
values of C_f are higher than before. The high inaccuracy of the old wall function is reflected also in the wall-tangent velocity profiles in Fig. 4.25a: the profiles of the cases with first cell thickness of y^+=25 and 50 present the wrongly predicted detachment discussed above. Furthermore also the profile of y^+=10 shows a relevant inconsistency with the finest solution. On the other hand the improved formulation is very accurate, showing full agreement with the finest resolution of the velocity profile.

Interesting to analyze is how the flow evolves along the airfoil's surface and how it influences the performance of the new wall-function boundary condition. Fig. 4.26 shows the velocity profiles scaled in wall units, at three different positions along the chord. The fit of the profile obtained without wall function to the theoretical curves (black lines) becomes progressively worse as the the flow advances towards the trailing edge. What happens is perfectly reasonable because proceeding towards the trailing edge the pressure gradient that the flow faces is more and more strong, thus the simplifying hypothesis made to obtain the linear and log law no longer fully apply. Consequence is also the worse fit between



(a) 25% of the chord.

(b) 50% of the chord.

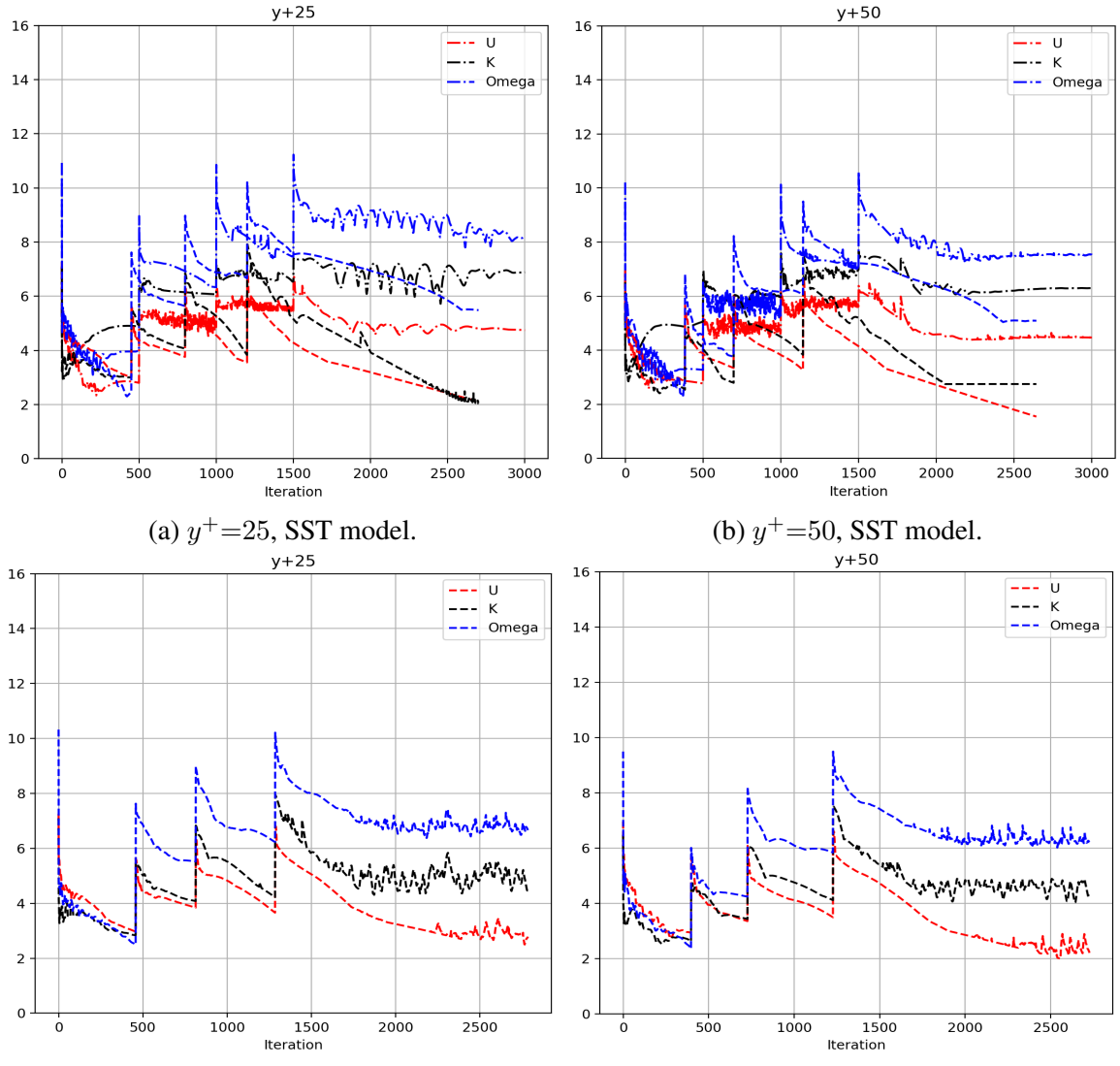


(c) 75% of the chord.

Figure 4.26 – Improved formulation, SST model: scaled wall-tangent velocity profiles at different positions along the chord, semilogarithmic axes.

the different y^+ cases, as we see in Fig. 4.26c. However it is positive to see that the resolution of the unscaled velocity profiles at the 75% of the chord is not affected at all, as Fig. 4.25b displays, having curves perfectly fitted to the reference solution for all the cases that implement the improved formulation on the coarser grids.

The poorer numerical stability of the previous formulation is confirmed by the presented results. In Fig. 4.27 the residuals' convergence with the previous formulation shows unstable oscillations absent in the improved formulation. Thus it's possible to state that the wrong prediction of the separation is influenced also by the poor numerical stability, besides the poor accuracy, of the old wall-function boundary condition. In Fig. 4.27 the same plots are reported also for the EARSM model. The trend is very similar to the SST model, with the exception of some oscillations of limited amplitude. These are probably caused by the higher



(a) y^+=25, SST model.

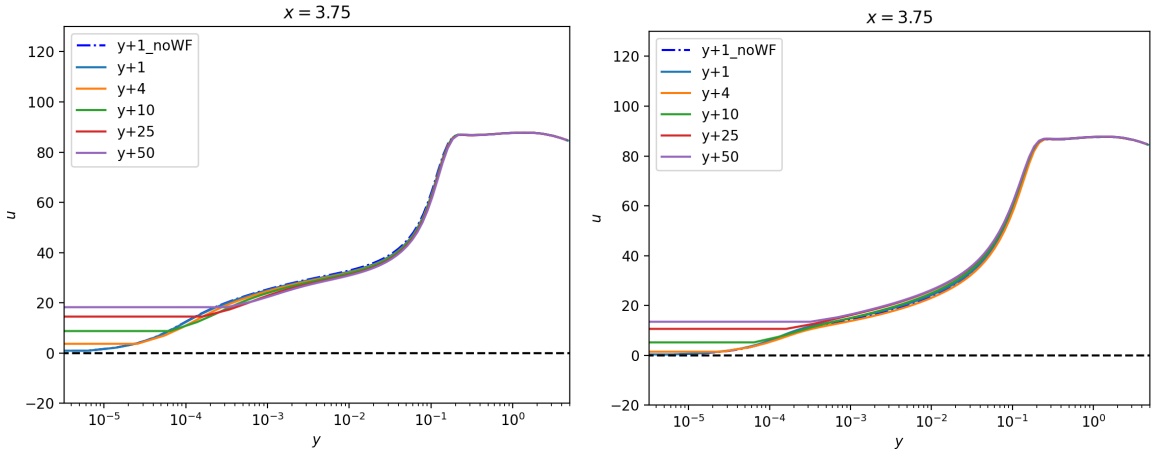
(b) y^+=50, SST model.

(c) y^+=25, EARSM model.

(d) y^+=50, EARSM model.

Figure 4.27 – Formulations comparison: convergence of the residuals, with old (dash-dotted lines) and new (dashed lines) wall function.

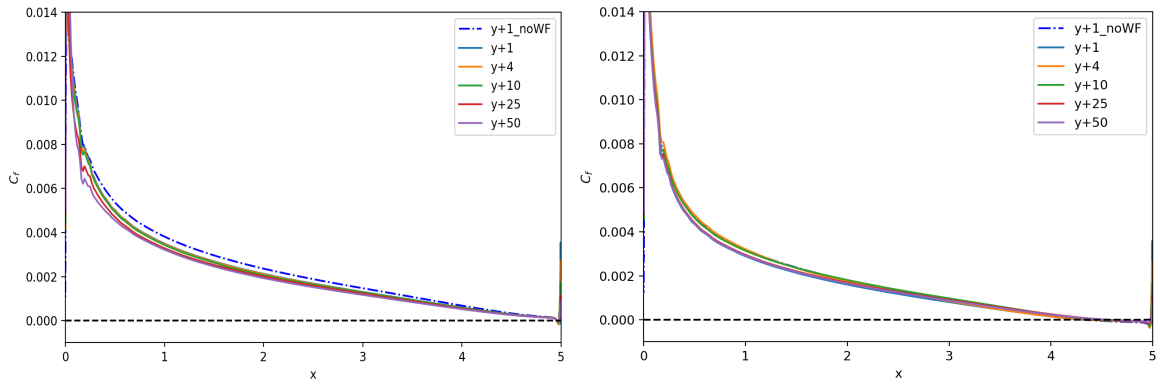
complexity of the model, which makes it a bit stiffer. Anyway such oscillations do not affect the convergence of the integral forces, which is smooth and the predicted values are in full accordance with the SST model. Also the resolution of the skin friction coefficient and the velocity profiles is practically the same as the SST model, which is always positive to observe as it means that the new wall function maintains the same good performance with different turbulence models.



(a) EARSM model. (b) SA model.

Figure 4.28 – Improved formulation: unscaled wall-tangent velocity profiles, semilogarithmic axes.

Equally positive to observe is the similarity between the results of the SA model and the $k-\omega$ models. An example of this are the velocity profiles at the 75% of the chord, showed in Fig. 4.28 for the EARSM and SA models. Like observed in the previous study cases, something that changes a bit in the SA model is the skin friction coefficient, predicting slightly higher values. The behaviour is confirmed by the plots in Fig. 4.29. In the same plots we can appreciate how, in this study case, for the SA model the gap between the reference solution



(a) EARSM model. (b) SA model.

Figure 4.29 – Improved formulation: skin friction coefficient on the airfoil surface with AOA=15°.

without wall functions and the others is not present any more. The higher skin friction values predicted do not affect the drag coefficient, since in the case here tested the major contribution comes from the pressure field around the body, which prediction is basically the same with all the turbulence models implemented. This last aspect is very remarkable as ensures the consistency in the prediction of lift and drag coefficients, usually the parameters of interest in similar flow cases, even when using different turbulence models.

4.3.2 Angle of attack = 18°

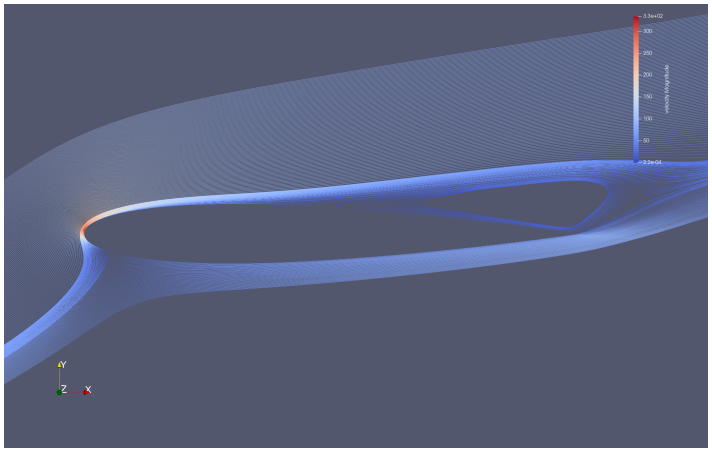


Figure 4.30 – AOA=18°: streamlines colored by velocity magnitude.

A consistent flow detachment is presented by the case with angle of attack equal to 18°. On the rear half of the upper surface of the wing profile a separation bubble of relevant dimensions appears, which can be observed in Fig. 4.30. It was preferred to non treat a fully separated flow, like in the stall condition, because it would represent a really unsteady condition, computationally more tricky to treat and not resolvable consistently with a steady state calculation. The bad performances of the old formulation highlighted in the previous test made useless to test it also for this case. Thus much focus was put into analyze the performance of the new formulation, particularly in the region covered by the separation bubble, and how it behaves with the three different turbulence models. Having negative values of skin friction the normalization in wall units looses its meaning and hence the analysis of scaled velocity profiles is avoided in this context.

According to Fig. 4.22, the separation of the flow should start in the proximity of the 40% of the chord, where the skin friction assumes a zero value. A further iterated (around 4000 for full convergence) simulations with the SST models and all the the different grids give the results shown in Fig. 4.31, which indicate the start of the separation a bit downstream. The match among the different curves is good in the region where the flow is attached, with some relevant difference only for the cases with y^+=25 and 50. Appreciable is how all the curves reach the zero value at the same point and come to an almost perfect agreement in the detached-flow area. This is an evidence of how the separation bubble, the feature of

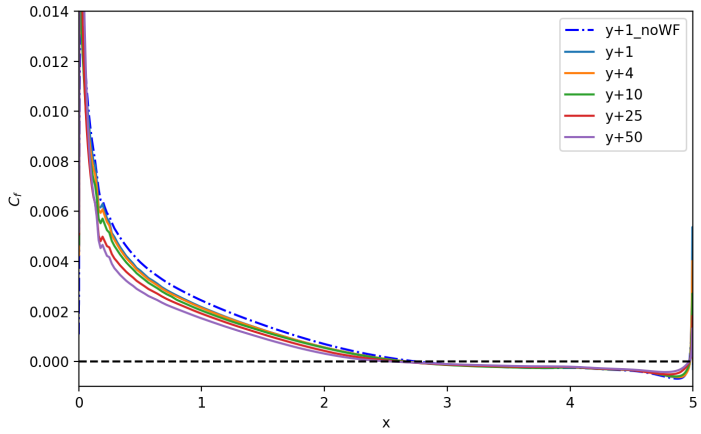
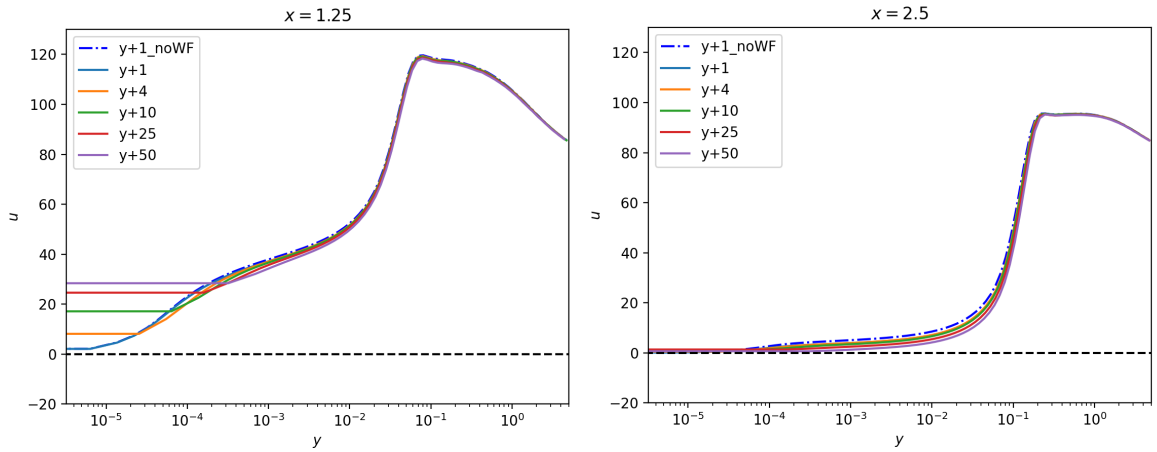
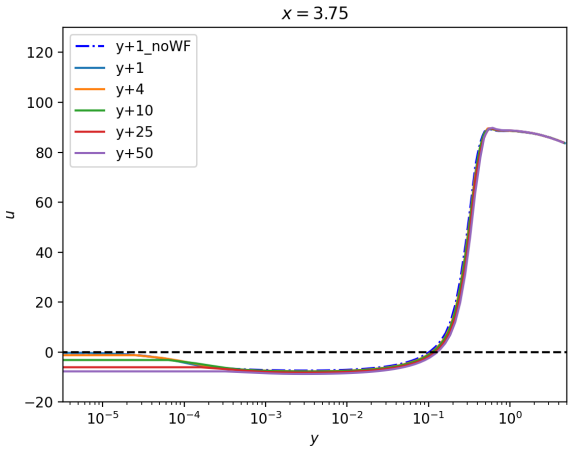


Figure 4.31 – Improved formulation, SST model: skin friction coefficient on the airfoil surface with AOA=18°.



(a) 25% of the chord.

(b) 50% of the chord.

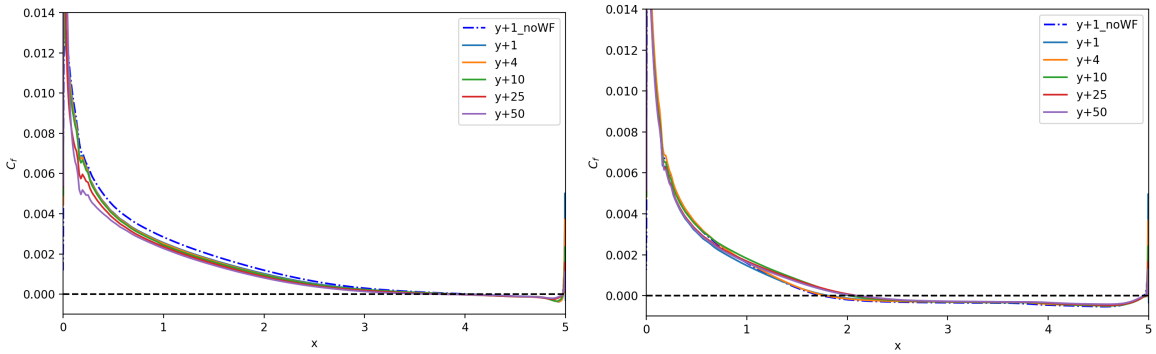


(c) 75% of the chord.

Figure 4.32 – Improved formulation, SST model: wall-tangent velocity profiles at different positions along the chord, semilogarithmic axes.

interest in this test, in equally resolved by all the cases, with and without wall function. Thus is possible to say that the improved formulation well performs in this situation.

In support of the previous claim, Fig. 4.32 shows the resolution of the unscaled velocity profiles at the three different positions. Such positions happen to be in three different flow situations, representing a very valid test: at the first one, the 25% of the chord, the flow is fully attached; at the second one, the 50% of the chord, the flow is very close to the zero skin friction point and the wall-normal derivative of the velocity is very close to zero; at the third one, the 75% of the chord, the flow is detached and the velocity presents negative values. In all the three cases the wall-function boundary condition gives proof of good performances, having very good agreement among all the cases.

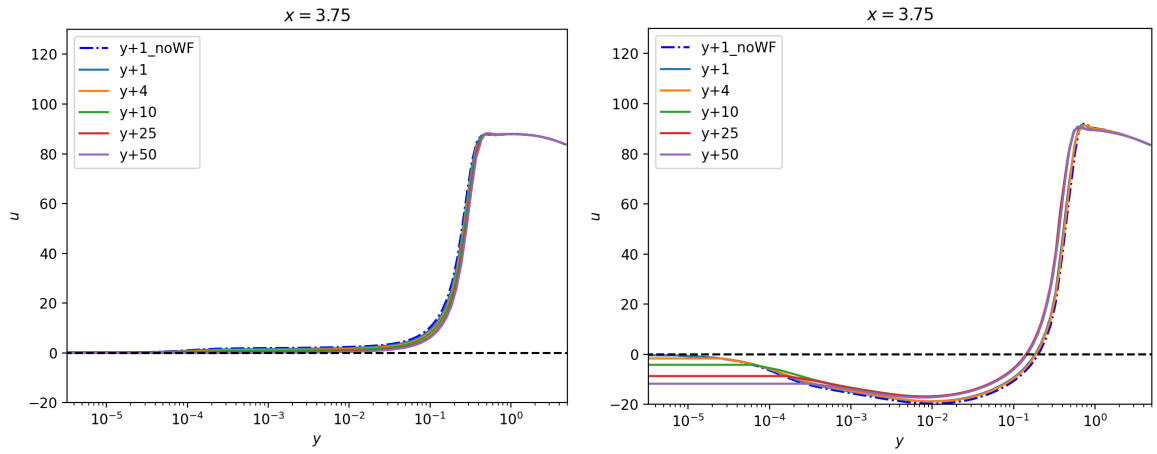


(a) EARSM model.

(b) SA model.

Figure 4.33 – Improved formulation: skin friction coefficient on the airfoil surface with AOA=18°.

The flow resolved by the other two turbulence models is slightly different. In Fig. 4.33 are reported the C_f curves for the EARSM and SA models. In both cases the point of zero skin friction is different from the SST case. Moreover this is something not related to the wall-function boundary condition because the different resolution is shown by the cases without wall function as well. What happens perfectly makes sense because RANS models resolves the phenomenon with a qualitatively good results and not quantitatively, i.e. the flow resolved is consistent with reality but is not ensured that corresponds to that specific angle of attack considered. Thus different models can give the flow corresponding in the reality to different angles of attack. Telling which is the correct one is not easy (more accurate turbulence models are needed) an is not the aim of the presented work. The focus here is on the performance of the improved formulation and it is remarkable in all the models. The different flow resolution leads to different conditions in the velocity profiles. In Fig. 4.34 the profiles at the 75% of the chord are presented. For the EARSM model, that predicts a postponed separation compared to the SST model, the profiles are close to a zero-derivative condition. With the SA model on the other hand, which predicts a slightly anticipated separation, the velocity presents larger negative values. To appreciate is the great match of the wall function cases with the reference solution. The only exceptions are the cases with y^+=25 and 50 for the SA model. Fig. 4.30 shows how the boundary layer around the leading edge is very thin, Thus with high y^+ at wall the first portion of boundary layer is not covered. Even a slightly different resolution at the

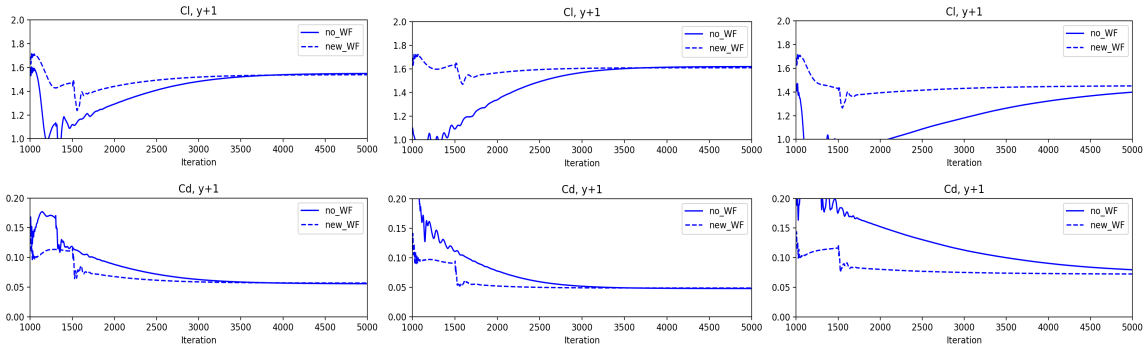


(a) EARSM model. (b) SA model.

Figure 4.34 – Improved formulation: wall-tangent velocity profiles, semilogarithmic axes.

beginning of the boundary layer can sensibly influence the whole development all along the wing profile, which can be the explanation for the small differences in the velocity profiles close to the end of the trailing edge. Moreover, the oscillations around the convergence value in the residuals suggest that also the velocity oscillates a bit around an average value that could not correspond to the one plotted, depending on the stop of the simulation. Anyway the difference is not so relevant and the accuracy of the prediction is still significant.

A different flow resolved means a different pressure field, which also means different forces on the body. The integral forces plots report indeed slightly different convergence values. In Fig. 4.35 the y^+=1 cases' convergence is presented. The coefficients of lift and drag change between the different turbulence models but all of them reach a smooth convergence with the improved formulation. Very valuable is that such convergence is faster with wall functions than without, as the plots showcase. Within the same turbulence model, the different y^+ cases find a consistency in the prediction of C_l and C_d with the finest solution. It is the natural consequence of a very good agreement in the resolution of the pressure field



(a) SST model. (b) EARSM model. (c) SA model.

Figure 4.35 – Improved formulation: convergence of the integral forces of y^+=1 case with different turbulence models.

around the body surface. The good performances of the new wall function formulations bring a sensible reduction in the computational time, with almost equal accuracy to the reference solution. In Table 4.3 is reported the CPU time (minutes per 1000 iterations) employed by the different cases simulated. The considerable reduction in the number of grid points from the y^+=1 to the the y^+=50 case leads to a sensible reduction in time. Moving from the finest solution without wall function to the highest first cell thickness results in a reduction in time of at least the 30%. Comparing the different turbulence models, the different time required reflects the complexity of the model and hence the possible accuracy, with the Spalart-Allmaras model being the simplest and having the lowest computational time. To

Table 4.3 – NACA 0012 case, AOA=18°: number of grid points and computational time (minutes per 1000 iterations) used by the different y^+ cases, when implementing the different turbulence models.

y^+	no WF	1	4	10	25	50
Number of grid points	23005	23005	20040	18041	16121	14636
SST	5.35	5.27	4.69	4.21	3.72	3.40
EARSM	5.69	5.51	4.83	4.36	3.82	3.54
SA	4.64	4.57	4.09	3.69	3.28	3.04

notice is also the reduction brought by using the improved wall function on the y^+=1 mesh. Considering that the accuracy of the solution is very close and that the solution with wall function converges faster (see Fig. 4.35), it is possible to assess that the new formulation not only has a great adaptability to different refinements of the near-wall grid but can also represent an improvement in very refined grid to reduce the computational time required by the simulation.

4.4 NACA 0012 Airfoil, Ma=0.8

The next test aims to introduce the disturbance of a shock wave in the flow. The 2D geometry of the NACA 0012 airfoil is retained and the free-stream conditions are the same as in the previous cases. The only free-stream quantity that is changed is the velocity, in order to have a Mach number correspondent to a compressible, subsonic regime. Setting Ma=0.8, the free-stream velocity obtained is $U_\infty = 278$ m/s and it is set with a small angle of attach equal to 3°. Consequently, keeping the same values of chord and free-stream viscosity as before, the Reynolds number increases to 88 millions. Since now the Mach number is high, the numerical preconditioning adopted for the tests with Ma≈0.2 is no longer implemented. All the other numerical setups are maintained the same. The boundary conditions are the same described in section 4.2. The mesh is the same as the previous cases for the unstructured part while the anisotropic layers in the near-wall region are re-generated according to the new free-stream velocity and Reynolds number. Having a larger Reynolds number, the first cell thickness corresponding to y^+=1 is smaller than previously. Moreover, retaining the same growth rate the number of layers required to cover the full boundary layer thickness increase and so does the overall number of grid points. The new set of meshes for this study case present around the 10% more of grid points. In the light of the limits presented in the previous tests, it is avoided to simulate the old formulation in this case. The focus is more on how the different turbulence model react to the disturbance of the shock wave. Below are presented and discussed the results obtained with the SST model for first, and then the others.

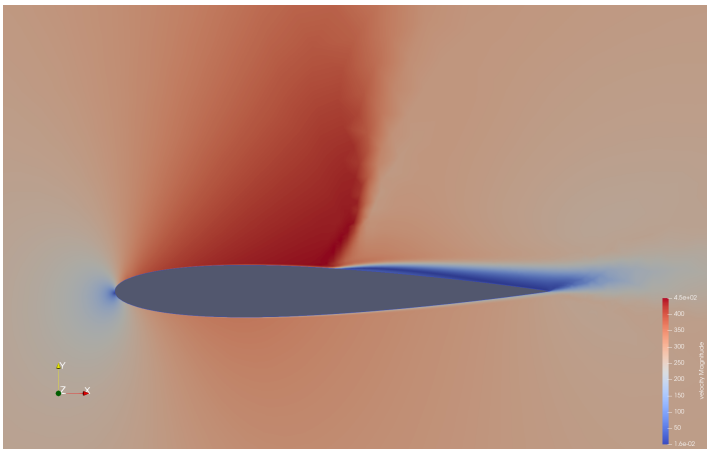
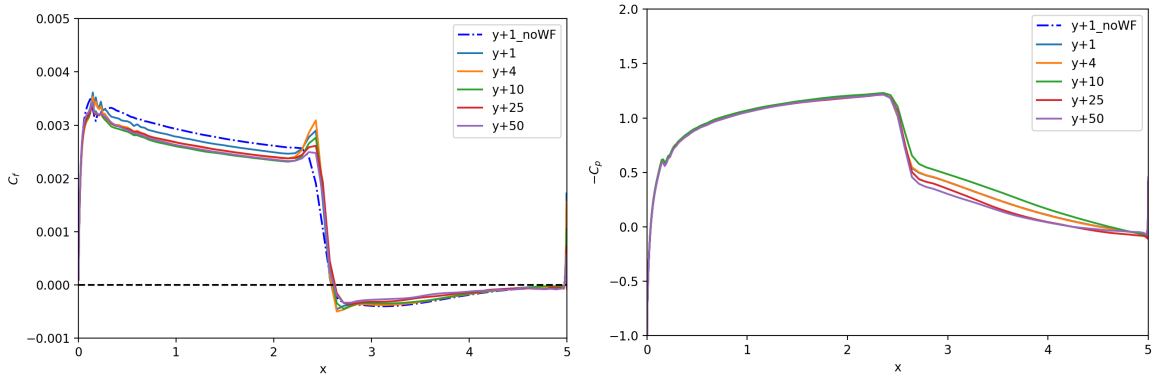


Figure 4.36 – Flow solution of the transonic airfoil case obtained with SST model and no wall function.

In Fig. 4.36 is presented the flow solution obtained on the finest grid with the SST model and no wall function. The colormap is based on the velocity magnitude and it allows to identify some characteristics of the flow evolution: the stagnation point is identified by the intense blue dot at the leading edge; on the upper surface of the wing profile the flow accelerates (more intense red tone) until the shock wave is formed close to the half of the chord and the velocity drops; after the shock wave the flow detaches from the body surface and separates in a blue wake.

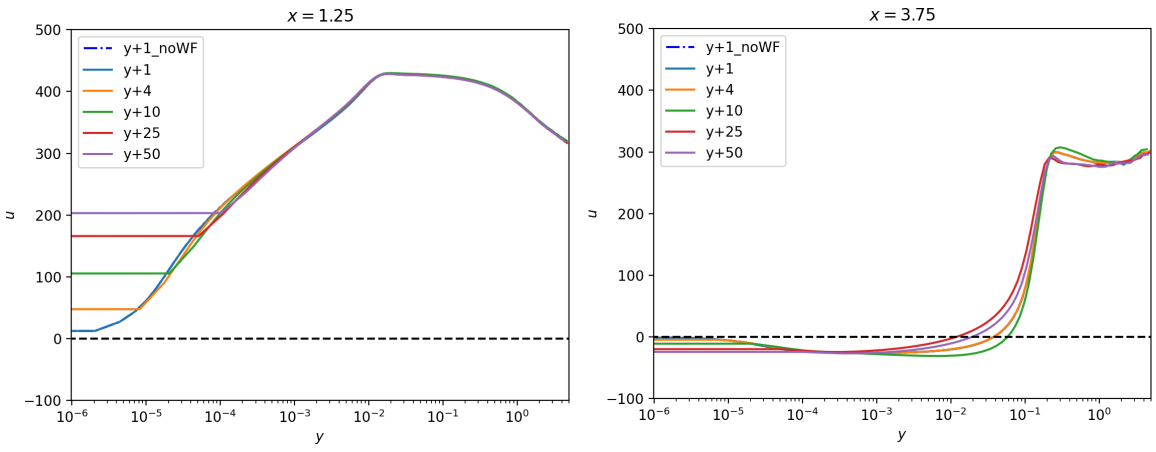
The presence of the shock wave can be recognized also in the skin friction coefficient and pressure coefficient distributions, reported in Fig. 4.37. Just like verified in the previous study



(a) Skin friction coefficient. (b) Pressure coefficient.

Figure 4.37 – Improved formulation, SST model: skin friction and pressure coefficient along the upper surface of the transonic airfoil.

cases, in the region where the flow is attached a very good agreement between the different cases is shown by the plots. The situation changes when it comes to the shock wave: in Fig. 4.37a it is possible to observe how all the cases with wall-function boundary condition present an unphysical peak in the skin friction coefficient that is absent in the resolution without wall function. We can guess that it is something related to the theoretical nature of the wall function and an incompatibility with very strong pressure gradients. Neverthless the prevision of the separation point (C_f<0) is the same as the reference solution but after that all the cases with y^+>4 show disagreement with the reference solution and the finer y^+ cases. The disagreement can be observed both in the pressure coefficient (Fig. 4.37b) and in the velocity profiles in the separation region (Fig. 4.38b). The origin point of the separation is a very

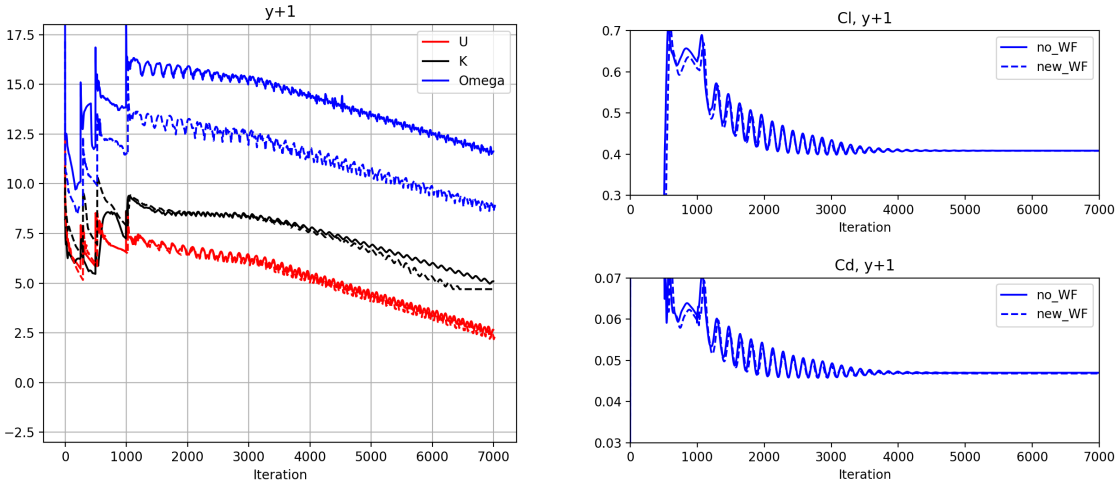


(a) Skin friction coefficient. (b) Pressure coefficient.

Figure 4.38 – Improved formulation, SST model: wall-tangent velocity profiles along the upper surface of the transonic airfoil.

sensible resolution and even small differences can lead to larger differences downstream, where the separation is more developed. On the other hand it is very good the agreement of the velocity profiles in the region upstream of the shock wave.

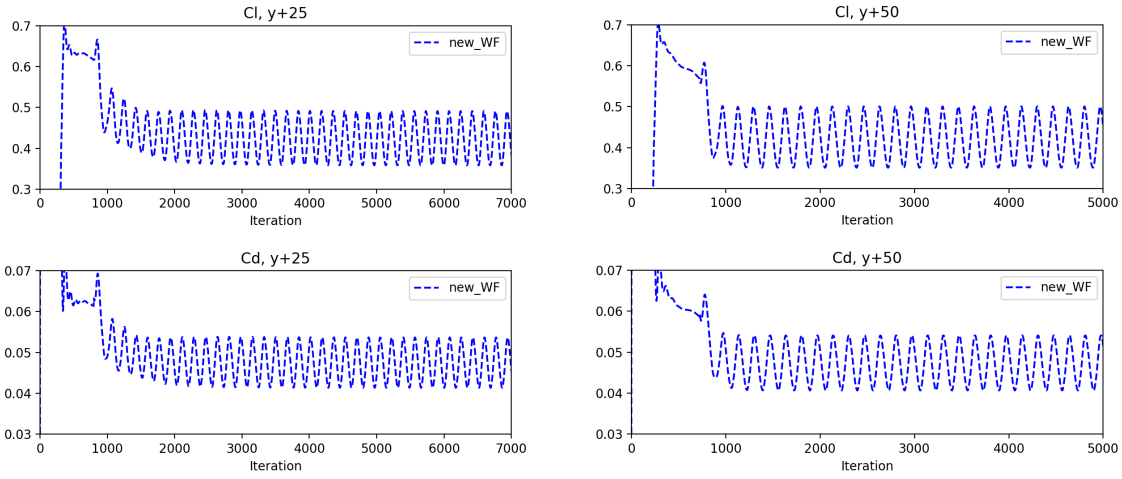
The disturbance of the shock wave significantly affects the convergence of the simulation. In Fig. 4.39 is displayed the convergence on the y^+=1 grid. The residuals constantly present oscillations, even if of small amplitude. The same oscillations are present in the integral forces' convergence but after 4000 iterations they level out to a constant value. The persistent



(a) Residuals. (b) Integral forces.

Figure 4.39 – Improved formulation, SST model: convergence of the y^+=1 case with no (solid line) and new (dashed line) wall function.

presence of oscillations suggests that the instability has not a numerical cause but a physical one. A possible cause can be that the physical condition that is simulated naturally presents some kind of vortex shedding or some large scale instability, like transonic buffeting. The

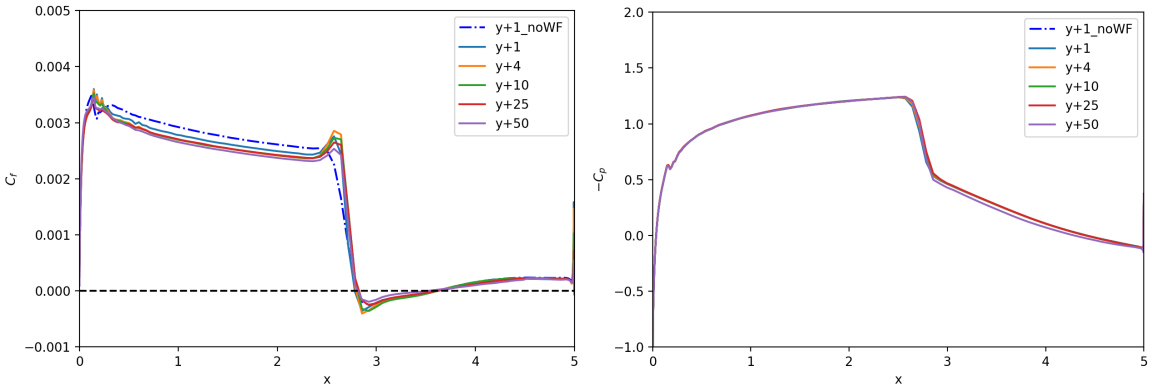


(a) $y^+ = 25$. (b) $y^+ = 50$.

Figure 4.40 – Improved formulation, SST model: convergence of the integral forces.

buffeting consists in a cyclical oscillation of the shock wave along the surface of the wing profile. It is something physically correct but very unsteady, thus represents a source of instability in a steady state computation. Also the convergence plots for the higher y^+ cases, reported in Fig. 4.40, support the hypothesis of a physical instability: the integral forces coefficients oscillate with constant amplitude around a constant value. Such value is slightly different from the one computed in the finest solution, in accordance to the fact that the region after the shock wave is solved slightly differently.

As expected, the flow resolved with the other turbulence models differs in some aspects with the SST model's solution. With the EARSM model, the trend of the C_f (Fig. 4.41a) has many analogies with what seen previously, but with two substantial differences. The first one is that the position of the shock wave is predicted a bit more towards the trailing edge. The second is that after a small separation the flow reattaches. The levels of accuracy and agreement are the same of the SST model in the attached region and at the shock wave, while are improved in the downstream region. In that region the pressure coefficient prediction (Fig. 4.41b) finds consistency also with the coarser meshes, with only a small gap of the y^+=50 case.

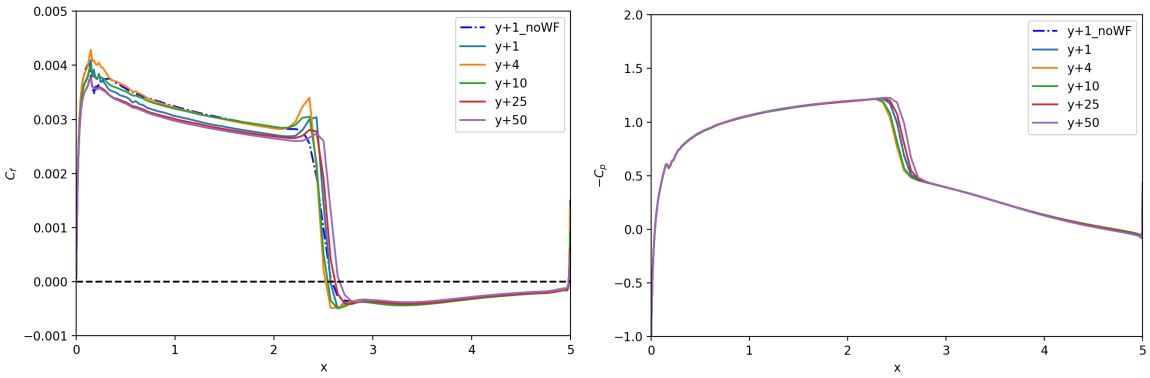


(a) Skin friction coefficient.

(b) Pressure coefficient.

Figure 4.41 – Improved formulation, EARSM model: trends along the upper surface of the transonic airfoil.

The flow resolved with the SA model, and without wall function, presents more or less the same characteristics as with the SST model: the shock wave is located close to the 50% of the chord and after that the flow is separated, without reattachment. Sensibly different is the behaviour of cases with wall function. The development of skin friction coefficient, in Fig. 4.42, does not have a very good agreement among the cases with wall function and between these and the reference solution, both in the region where the flow is attached and where the shock wave is. Surprisingly, in the region where the flow is detached all the cases show a perfect match. This is showed also by the pressure coefficient development in Fig. 4.42, having a perfect agreement before and after the shock wave, but not in the proximity of it. One important fact to be noted is how all the models present peaks of C_f immediately before and after the shock wave, when implementing the wall functions. Such peaks are larger for the more refined grids and, even if the resolution of the shock wave is not compromised, they



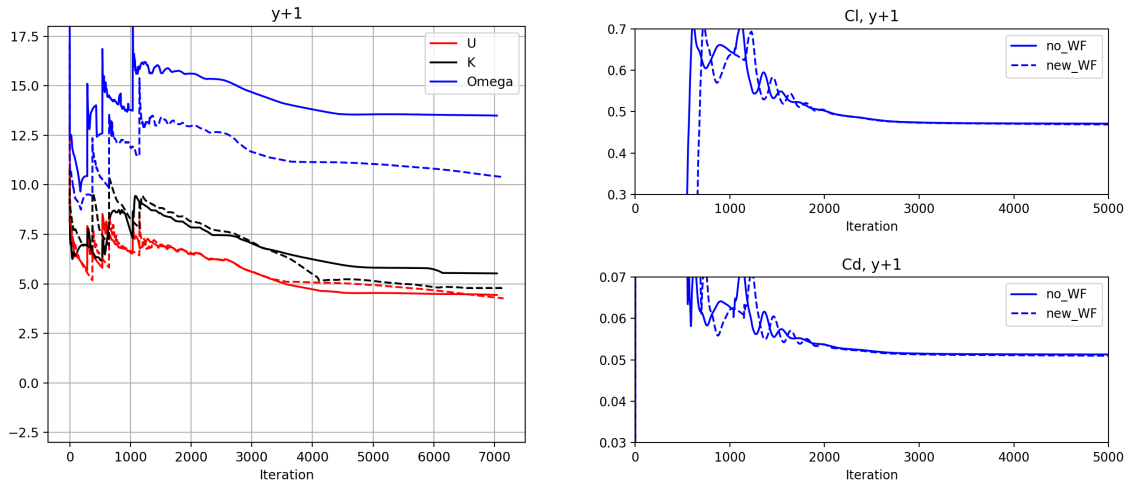
(a) Skin friction coefficient.

(b) Pressure coefficient.

Figure 4.42 – Improved formulation, SA model: trends along the upper surface of the transonic airfoil.

show difficulties of the improved formulation to handle such strong gradients.

The oscillations present in the convergence for the SST model are almost absent in the finest solution for the EARSM (Fig. 4.43a) and SA (Fig. 4.44a) models. For these models, also the cases with higher y^+ do not present relevant oscillations, with the force coefficients smoothly leveled out. An hypothesis can be that either the flow simulated is close to have unsteady phenomena, but not exactly in that situation, or the nature of the EARSM and SA models dump such unsteady phenomena. Comparing the integral forces' convergence for the three models (Fig.s 4.39b, 4.43b and 4.44b), one common aspect is that, contrary to the previous tests, the implementation of the wall-function boundary condition doesn't provide any improvement to the convergence of the case with y^+=1. Thus, if previously it was a good option to apply wall functions also to the finest grids, now it's not anymore, as the accuracy

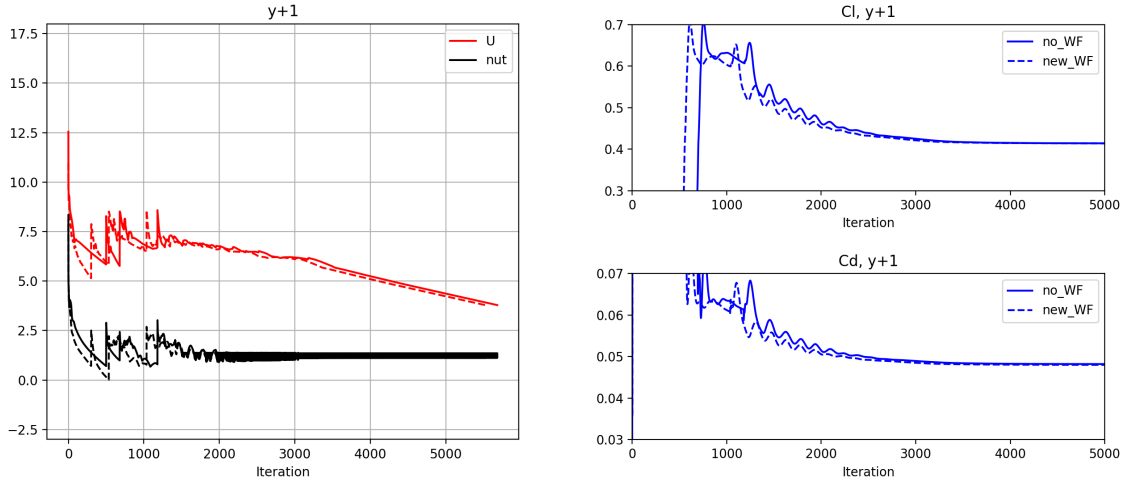


(a) Residuals.

(b) Integral forces.

Figure 4.43 – Improved formulation, EARSM model: convergence of the y^+=1 case with no (solid line) and new (dashed line) wall function.

of the skin friction coefficient is compromised. Looking at the same plots, to observe is how with the EARSM model the lift and drag coefficients are slightly larger. This is the logical consequence of the flow reattachment, which causes in that region both an increase of the pressure coefficient and of the skin friction.



(a) Residuals.

(b) Integral forces.

Figure 4.44 – Improved formulation, SA model: convergence of the y^+=1 case with no (solid line) and new (dashed line) wall function.

4.5 ESAV Generic Fighter

The geometry to be used in the presented test is an Efficient Supersonic Air Vehicle (ESAV) generic fighter. The model has no official origin but was chosen as already cleaned up and trimmed appropriately to be meshed in Pointwise. In fact the no particular issue was faced in the meshing process but so wasn't for obtaining a barely valid solution from the simulation. It was very problematic to reach convergence in the simulations, both for the complexity of a 3D case and also for the strong characteristics of the model. The suspect is in fact that, being not official, the model is pretty schematic, without a proper definition of the wing profiles all along the wingspan. The wing close to the tip is indeed extremely thin and probably unrealistic. Such conclusions were driven when already a significant effort was put into the test, thus the study case was concluded and the results are presented, without claiming to be fully correct but only as a cue for some considerations. The model and the computational domain are displayed in Fig. 4.45 and the principal dimensions of the geometry are reported in Table 4.4. The model is characterized by sharp edges and the presence of the intake and

(a) Body. (b) Far-field region.

Figure 4.45 – ESAV case: model.

Table 4.4 – ESAV case: model dimensions.

Length	19.50	m
Wingspan	17.28	m
Root chord	6.86	m
Tip chord	1.37	m
Reference chord	3.38	m
Reference surface	29.16	m^2

the nozzles of the engine. The free-stream conditions are kept the same as before, having a compressible subsonic regime with Ma=0.8. The free-stream velocity is still equal to 278 m/s with an angle of attack of 3° and the Reynolds number computed with the reference

chord is around 60 millions. In order to save grid points and thus computational expense, the model consists only in half of the plane. As a consequence a symmetry plane was created connecting the cut border of the body to the far-field boundary. The far-field boundary consists in a hemisphere and is placed around 60 reference chords away from the body. On the body surface an adiabatic wall boundary condition is retained. For the supersonic intake a pressure-outlet boundary condition is selected, prescribing on the boundary surface the static pressure obtained by isentropic expansion from Ma=0.8 (free-stream value) to Ma=0.9, estimating this way the effect of the convergent section of the intake. At the nozzle a total states inlet boundary condition is set. The overall effect of the engine is estimated in an increase of the 20% in the total temperature and of the 5% in the total pressure. The values obtained in this way are prescribed at the boundary surface. The surface mesh generated on the fighter's body (Fig. 4.46a) is of unstructured type and counts 55870 grid nodes. The unstructured mesh type is adopted, with refined density in the proximity of corners and edges. Anisotropic layers are used to control the cell size and better resolve the curvature at the edges and corners of the model. Then a block is formed enclosed by the body surface, the symmetry plane and the

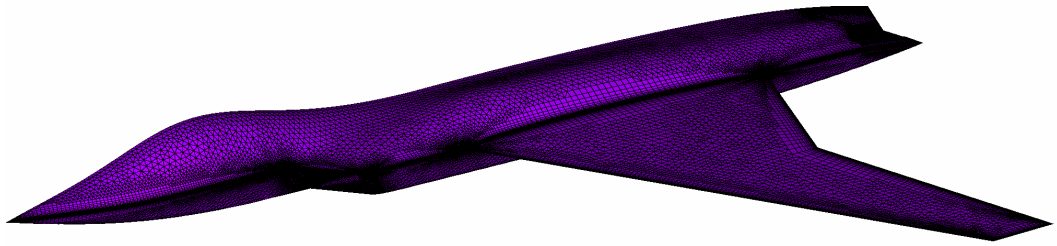


Figure 4.46 – ESAV case: surface mesh.

far-field hemisphere, and the volume mesh is generated setting parameters to have anisotropic layers with the desired first cell thickness all over the body surface. In Fig. 4.47 the grid on the symmetry plane is reported as an indication of how the volume mesh looks like, where is also observable the extrusion of anisotropic layers. For the presented study case two meshes are adopted: one with first cell thickness at $y^+=1$ and one at $y^+=25$. The first counts around 3.5 millions of grid nodes and the second around 2.4 millions.

As anticipated, the convergence of the simulations hasn't been an easy task as in the previous tests. Numerous attempts changing different solver's parameters were performed, following the suggestions of the M-Edge user manual, but acceptable solutions have been obtained only with one combination of changes. The computational parameters modified in such process are:

- CFL - the CFL number, which influences the local time step in Eq. 3.29, is lowered to 1.0 (default 1.5);
- NGRID - the number of grid levels in the multigrid strategy, is lowered to 2 (default 4);

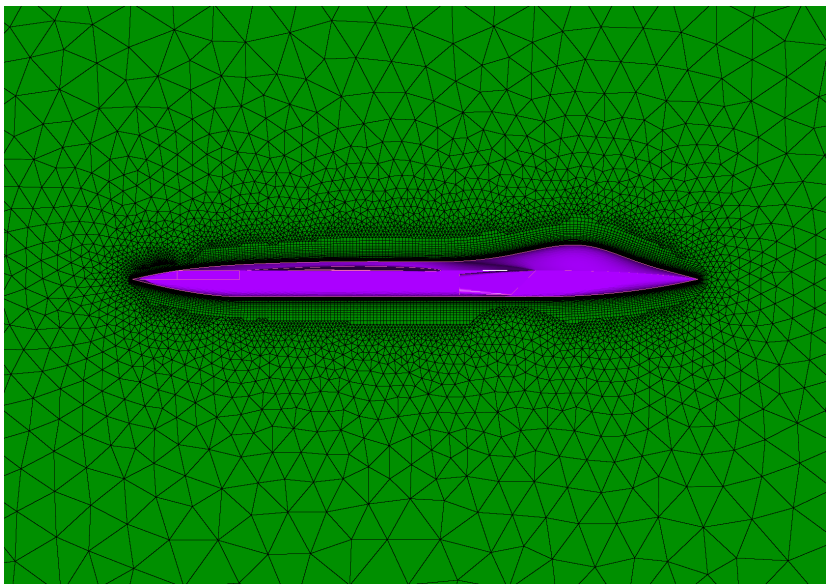
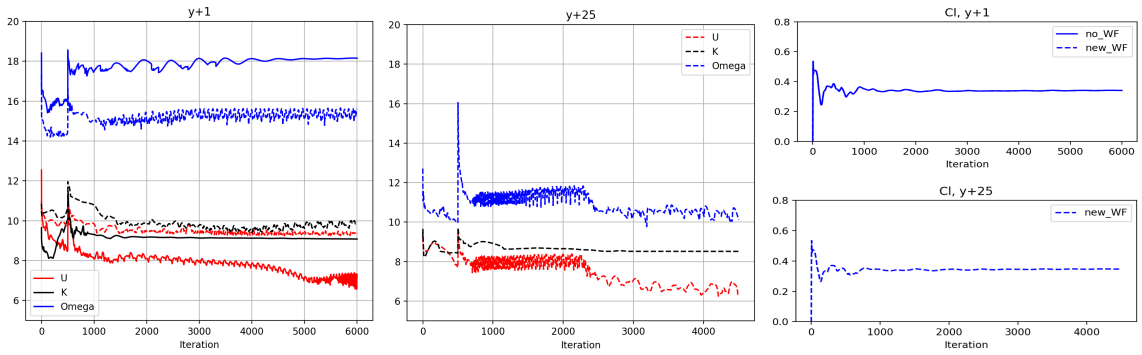


Figure 4.47 – ESAV case: volume mesh.

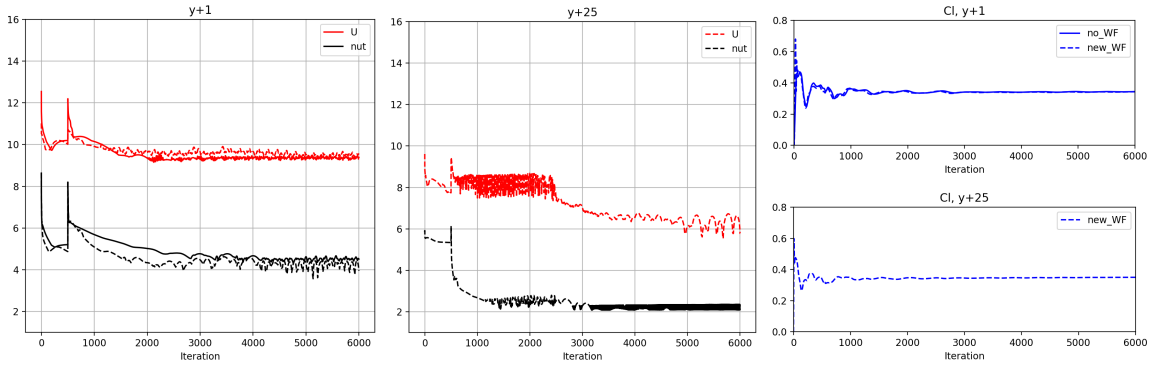
- VIS4 - corresponds to the constant $\kappa^{(4)}$, which controls the fourth order dissipation in Eq. 3.16. It is raised to 0.03 (default 0.02);
- TURFIX - is the parameter that controls the entropy fix for the turbulent equations and is set to -1.2 (default -1.1).

With the above adjustments acceptable results have been obtained for the SST and the SA models and are presented below. The convergence for the SST model can be analysed in Fig. 4.48. It is hard to say that the convergence is qualitatively good but at least presents some stability in the trend of the residuals, without the exponential growths faced in other attempts. The oscillations present in the residuals plots are a symptom of the numerical instability of the computation and to notice is that they are smaller in amplitude without the wall-function boundary condition. Positive is that such oscillations are not reflected in the convergence of the lift coefficient, that is presented smooth and levelled out for both the grid cases. Somehow



(a) Residuals, $y^+ = 1$. (b) Residuals, $y^+ = 25$. (c) Lift ceofficient.

Figure 4.48 – ESAV case: convergence for the SST model.

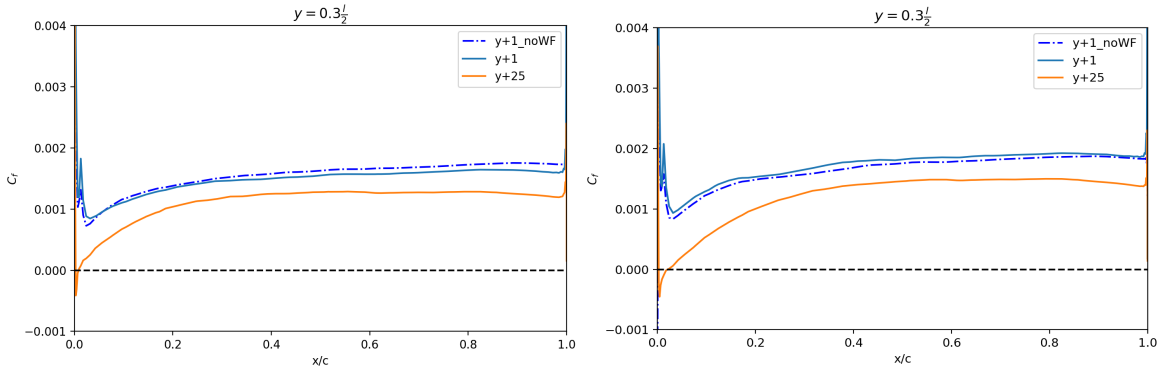


(a) Residuals, $y^+ = 1$. (b) Residuals, $y^+ = 25$. (c) Lift ceofficient.

Figure 4.49 – ESAV case: convergence for SA model.

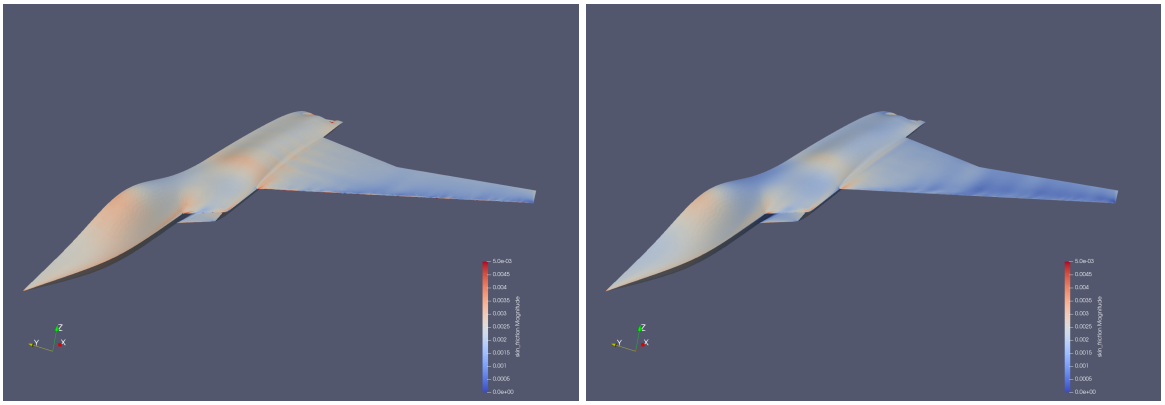
better is the convergence of the SA model, presented in Fig. 4.49. The residuals now show a larger reduction in the magnitude than before and get closer to a flattened development. Also here the solutions without wall-function boundary condition seem to be more stable than the ones with it.

The data for the plots are extracted from three different positions along the wingspan (y-direction) and also from one position on the fuselage. In all the characteristics analyzed the two turbulence models present very similar behaviours, affected by the same issues. Looking at the skin friction development along a wing's section, the instability is immediately clear. Fig. 4.50 shows the skin friction coefficient on the upper surface of the wing profile obtained sectioning the wing with a plane normal to the y-direction at the 30% of the half wingspan, i.e. close to the wing root. The big instability is at the leading edge, where very high peaks are presented by the cases with y^+=1. In the cases with y^+=25 the instability leads to the formation of a small separation bubble in the very beginning. To notice is that, with the exception of the initial offset, C_f presents the same trend for the two different grids. The difference in the prediction of the skin friction coefficient is not strictly limited to the wing.



(a) SST model. (b) SA model.

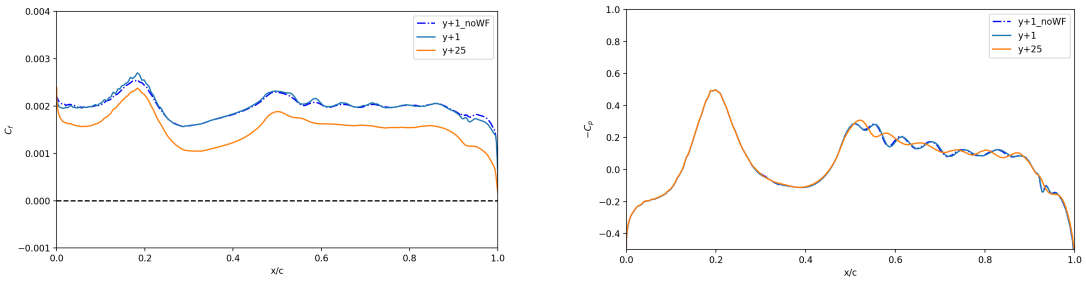
Figure 4.50 – ESAV case: development of the skin friction coefficient along a section of the wing.



(a) $y^+ = 1$. (b) $y^+ = 25$.

Figure 4.51 – ESAV case, SA model: surface map of the skin friction coefficient on the body.

Fig. 4.51 shows that a substantial difference persists in the majority of the regions on the body surface. The fact that the fuselage doesn't present such sharp and thin geometry as the wing would suggest that the difference should be absent, or at least reduced. This supposition is supported also by Fig. 4.52 where no instabilities are presented at the edges. The wiggles presented in both plots of the figure are not physical but caused by a bed resolution of the curved surface by the triangular cells of the surface mesh; the pressure at wall is very sensitive to the derivative of the surface, thus even small wiggles on the surface are amplified in the pressure at wall. The strong peaks, as well as the small separation, are not present in the C_f curves, however a relevant gap still persists between the different y^+ cases. The more stable condition on the fuselage leading edge is confirmed by the development of the pressure coefficient (Fig. 4.52b) where all the cases have a perfect agreement in the forward region. For sure the unstable flow on the wing influences the rest of the domain, included the region

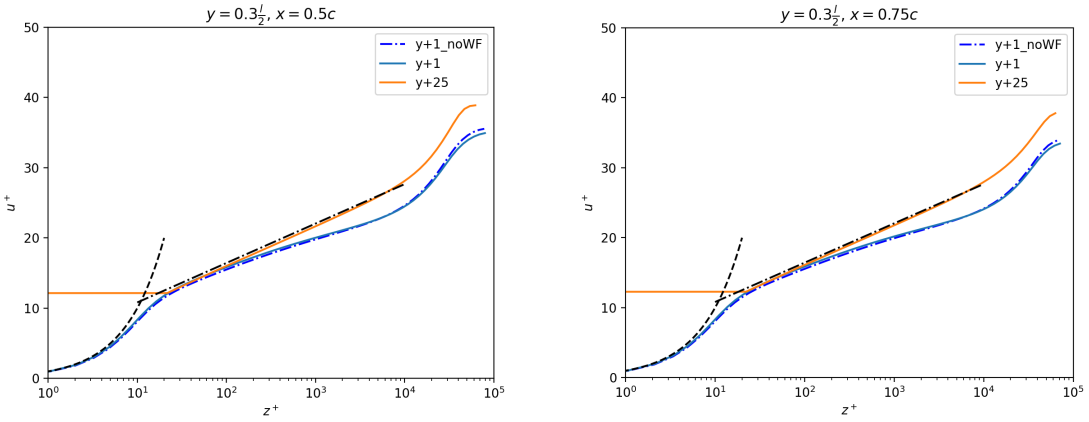


(a) Skin friction coefficient. (b) Pressure coefficient.

Figure 4.52 – ESAV case, SST model: trends along the fuselage

around the fuselage, but the incoherence described above pushes to question the efficacy of the improved boundary condition in this situation. On the other hand, many aspects suggest that such bad behaviour is not to be attributed to the performance of the wall function, first of all because in all plots the predictions with y^+=1 and wall function are very consistent with the reference solution. Another evidence can be found looking at the scaled velocity profiles on the same section of the wing, showed in Fig. 4.53. There is possible to observe

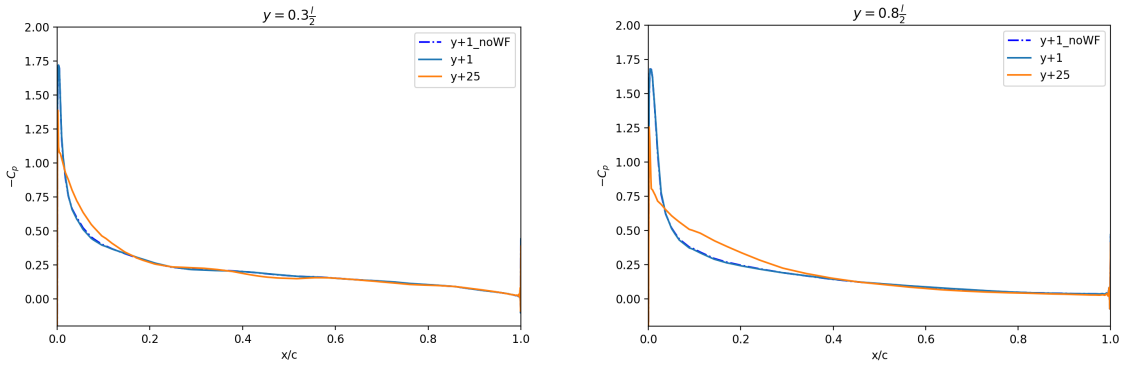
how, in both curves of the y^+=25 case, the first value of the scaled velocity is computed and set correctly by the boundary condition, i.e. the kink of the curve correctly lays on the y^+=1 profile. It should be observed the good agreement of the reference solution with the linear



(a) 50% of c. (b) 75% of c.

Figure 4.53 – ESAV case, SA model: scaled velocity profiles at two different positions, along a section of the wing.

law and with the log law, but only for a small region with the second. The increasing drift of the reference solution from the theoretical prediction as the distance from the wall increases is sign of the inadequacy of the log law for the simulated flow. Many factors are in fact in disagreement with the assumptions made by the law, first of all the fact that the flow is 3D and thus more components and effects are involved. The instability that affects the skin friction has of course an influence also on the pressure coefficient. In Fig. 4.54 the instability at the leading edge of the wing causes a considerable mismatch between the finer and coarser case. The same error is notably increased in the position at a higher percentage of the half wingspan, proving the fact that the instability at the leading edge increases advancing towards the wing tip, which can also be observed in Fig. 4.51.



(a) 30% of half wingspan. (b) 80% of half wingspan.

Figure 4.54 – ESAV case, SST model: development of the pressure coefficient along different sections of the wing.

Chapter 5

Concluding Remarks

The project presented described the development of an improved formulation of wall-function boundary condition to be implemented in the CFD solver M-Edge. Together with an improvement of the already existing formulation for $k - \omega$ models, a formulation for the Spalart-Allmaras models has been developed and tested. Various study cases, with increasing complexity, have been considered in order to explore individually different aspects of the flows that can influence the performance of a wall function. Once assessed the improvements of the new formulation and the weaknesses of the previous one, the tests continued only on the improved wall function, to deeper understand its behaviours and limits.

The improved formulation differs from the previous in some adjustments in the expressions of the formulae and in some details of the implementation of the boundary condition. Both these aspect gave evidence to have a positive effect on the overall performance of the boundary condition. Improvements brought by the new formulation have been very clear already from the first flow case tested, a 2D zero pressure-gradient flat plate. An evident improvement in the accuracy of the skin friction coefficient has been reported, which was more relevant for the solutions on grids with first cell thickness at high y^+ values. Keeping in mind that the main purpose of wall-function boundary conditions is to have less strict grid requirements without loosing accuracy in the near-wall flow, the first results of the test were already very remarkable. At this stage was reported also a considerable improvement in the numerical stability of the simulation, with a deeper and faster convergence of the solution. In the following airfoil test such improvements have been confirmed. Particularly interesting was to observe how the new formulation better handled the discontinuity of a sharp trailing edge.
The previous formulation showed more severe limits when a situation with flow detachment was simulated. Again the solutions with large y^+ at wall were the ones more compromised, with a complete wrong prediction of the near-wall flow. On the other hand the results of the improved formulation reported a remarkably good accuracy in the solution, also on the high y^+ grids which presented perfect agreement with the reference solution in the prediction of the lift and drag coefficients. Very positive was to observe how the good performance of the improved formulation was kept also when implementing different turbulence model, all with very similar results.
The improved formulation presented some difficulties when handling the strong gradients

introduced by a shock wave. These difficulties appeared in the form of unphysical peaks in the skin friction coefficient and a reduce stability in the convergence of the solution. Anyway these limits are more related to the nature of the formulation than to its adaptation to differently refined meshes. In fact all the grid cases and all the turbulence models presented such alterations.

In conclusion, the purpose of the presented work can be considered achieved: the new formulation sensibly improves the level of accuracy and the numerical stability of the wall-function boundary condition. These concretely translate in the possibility of relaxing the mesh requirements for a certain level of flow accuracy, with a consistent reduction of the number of grid point and consequently of the computational time. The largest y^+ value at wall considered in this work was 50. Compared to the finest solution with $y^+=1$, such case presented a reduction of almost the 40% in the number of grid points and a consequent reduction of more than the 35% in the CPU time required. With wall-function boundary condition values of y^+ even larger of 50 can be adopted, as long as the log law is consistent. This means that the reduction in computational expense can be further reduced.
Further investigation on the wall function's performance should be conducted on 3D cases. The choice of a complex and not so accurate 3D model in this work left open some questions and doubts in the efficacy of the wall function in full and complex 3D cases. Thus some tests on standard and extensively studied 3D flows would clear such unknowns. The formulation itself may be further improved to be able of handle the severe gradients generated by a shock wave. On option can be to introduce in the formulation some kind of limiter or artificial dissipation in order to eliminate those peaks of the skin friction coefficient in the vicinity of the shock wave, presented in the results of this project. Another possible development is a wall-function treatment to be implemented with full Reynolds stress models, to have the flow accuracy of this kind of models at a reduced computational expense.

References

[1] S. B. Pope, *Turbulent Flows*. Cambridge University Press, 2000. ISBN 9780511840531

[2] D. B. Spalding, "Monograph on turbulent boundary layer," *Technical report TWF/TN/33*, 1967.

[3] S. V. Patankar and D. B. Spalding, *Heat and Mass Transfer in Boundary Layers*. Morgan-Grampian Press, London, 1967.

[4] B. E. Launder and D. B. Spalding, "The numerical computation of turbulent flows," *Computer Methods in Applied Mechanics and Engineering*, vol. 3 (1974), pp. 269–289, 1973.

[5] C. C. Chieng and B. E. Launder, "On the calculation of turbulent heat transfer downstream from abrupt pipe expansion," *Numerical Heat Transfer*, vol. 3, pp. 189–207, 1980.

[6] R. W. Johnson and B. E. Launder, "Discussion of "on the calculation of turbulent heat transfer downstream from an abrupt pipe expansion"," *Numerical Heat Transfer*, vol. 5, pp. 493–496, 1980.

[7] M. Ciofalo and M. W. Collins, "$k - \varepsilon$ predictions of heat transfer in turbulent recirculating flows using an improved wall treatment," *Numerical Heat Transfer*, vol. Part B: Foundamentals, 15:1, pp. 21–47, 1989.

[8] T. Rung, "Universal wall-boundary conditions for turbulence-transport models," *Institutsbericht*, vol. 02/99, 1999.

[9] T. J. Craft, S. E. Gant, H. Iacovides, and B. E. Launder, "A new wall function strategy for complex turbulent flows," *Numerical Heat Transfer*, vol. Part B: Foundamentals, 45:4, pp. 301–318, 2004.

[10] D. Morgenweck, "Investigation and validation of a hybrid wall boundary condition in turbulent flows," Master's thesis, TU Berlin, 2005.

[11] P. Spalart and S. R. Allmaras, "A one-equation turbulence model for aerodynamic flows," *La recherche Aerospatiale*, vol. 1, pp. 5–21, 1994.

[12] D. C. Wilcox, "Turbulence modelling for cfd," *La Canada, CA: DCW Industries*, 1993.

[13] F. R. Menter, “Two-equation eddy-viscosity turbulence models for engineering applications,” *AIAA Journal*, vol. 32(8), pp. 1598–1605, 1994.

[14] S. Wallin and A. V. Johanson, “An explicit algebraic reynolds stress model for incompressible and compressible turbulent flows,” *J. Fluid Mech*, vol. 403, pp. 89–132, 2000.

[15] S. Wallin, “Improved wall function definition,” 2021, private communication.

[16] P. Eliasson, “Edge, a navier-stokes solver for unstructured grids,” in *Proceedings of Finite Volumes for Complex Applications III*, L. Hermes Penton Ltd, Ed., 2002. ISBN 1-9039-9634-1

[17] Turbulence modelling resource. Langley research center. [Online]. Available: https://turbmodels.larc.nasa.gov

Appendix A

Study of the flow separation for the NACA 0012

Below are presented the plots produced while seeking for angle of attack that presented interesting features to discuss the flow separation and the capability of wall functions to treat such flows. To check the presence of a separated flow the skin friction coefficient and the velocity field have been monitored, looking for negative values for the first and the presence of a separation bubble for the second. To sum up the study the $C_l - \alpha$ curve was built with the collected data, which is reported in Fig. A.3.

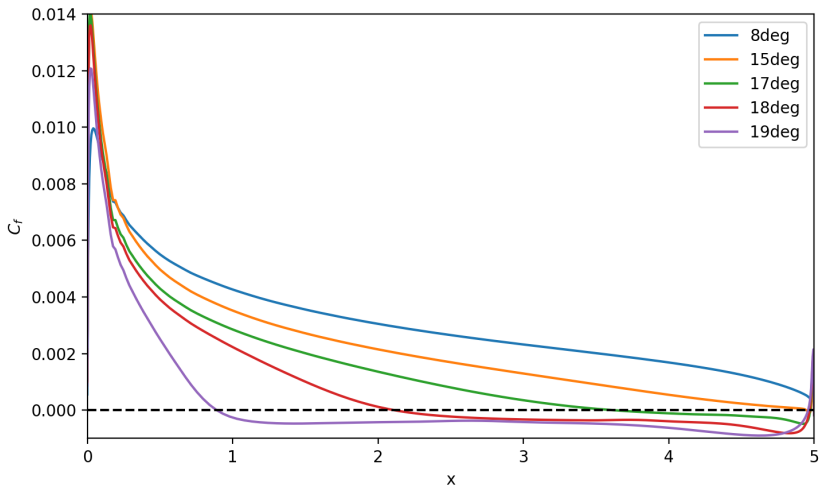
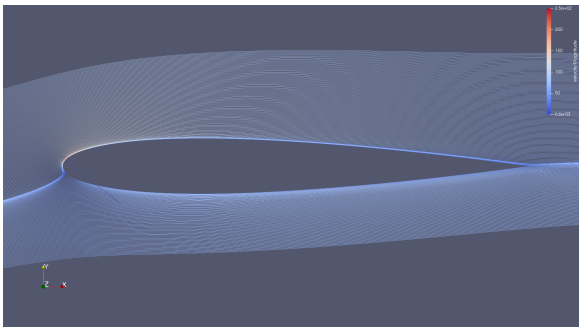
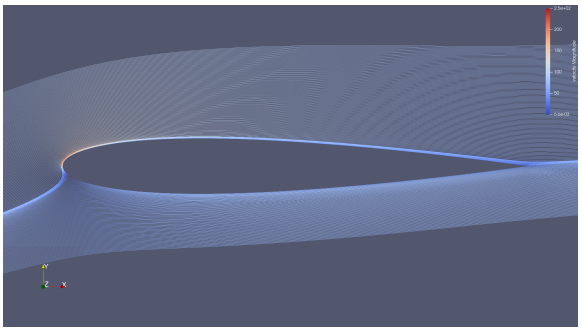


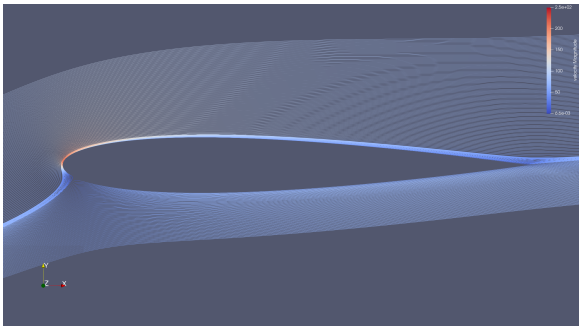
Figure A.1 – Flow separation study: skin friction coefficient along the airfoil.



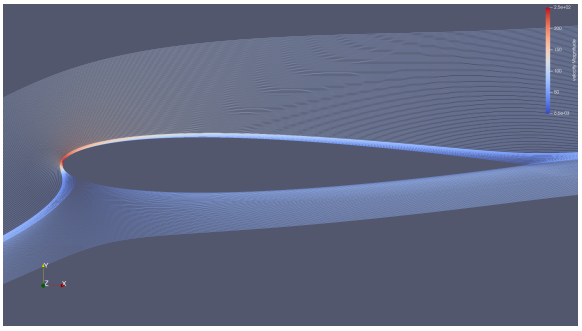
(a) AOA = 6°.



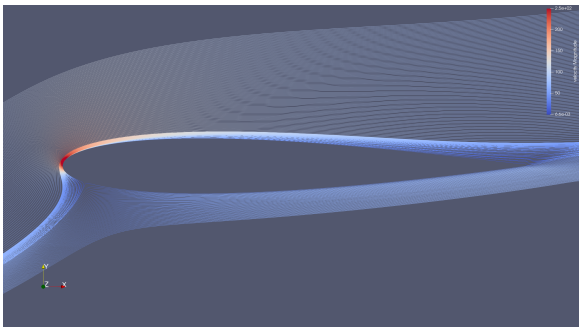
(b) AOA = 8°.



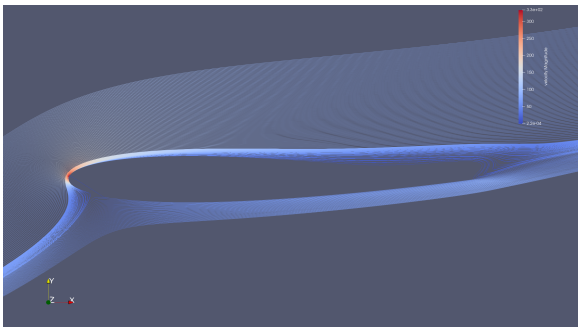
(c) AOA = 10°.



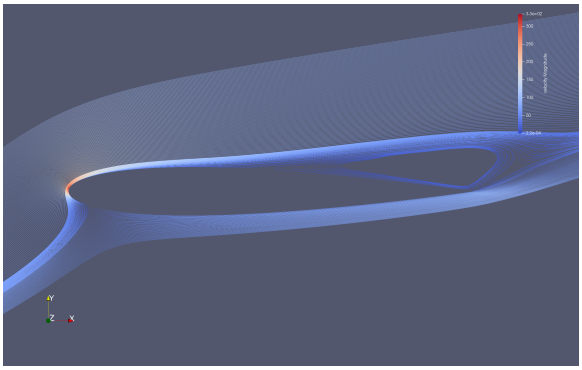
(d) AOA = 12°.



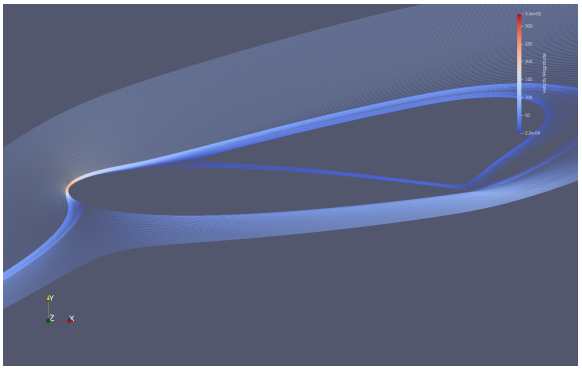
(e) AOA = 15°.



(f) AOA = 17°.



(g) AOA = 18°.



(h) AOA = 19°.

Figure A.2 – Flow separation study: streamlines in the velocity-magnitude field.

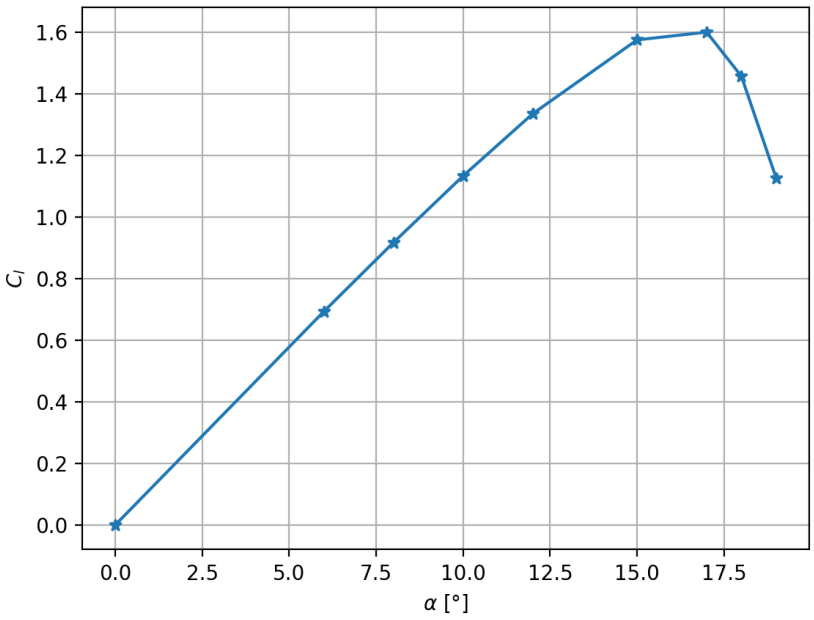


Figure A.3 – Flow separation study: $C_l - \alpha$ curve generated with collected data.

Acknowledgments

My most heartfelt thanks goes to my supervisor prof. Stefan Wallin. He has always been kindly available and extremely helpful throughout the development of my thesis project. The knowledge that I have acquired during this work, with his special contribution, will be precious for the continuation of my career.
I wold like to thank Faranggis Bagheri, Alessandro Porcarelli and Neptech AB for the support provided with Pointwise. A thanks goes also to Matteo Montecchia for assisting me in my first steps with M-Edge.
A special thanks goes to prof. Domenic D'Ambrosio for supervising me at my home university. Many thanks go to the Politecnico of Turin and the KTH university that made possible for me to live this amazing experience of exchange, which gave me so much from both the personal and academic point of view.

Il mio più sentito ringraziamento va ai miei genitori e alla mia famiglia. La persona che sono oggi e gli obbiettivi che ho raggiunto sono stati possibili solo grazie al loro supporto che mi ha permesso di non precludere mai i miei desideri e le mie aspirazioni.
Un sincero grazie va agli amici di una vita, persone speciali e insostituibili. Sono entrati a far parte della mia vita durante tutta la mia crescita e, nonostante il passare degli anni e le distanze fisiche presentatesi, ne continuano a far parte ogni giorno e non potrei farne a meno.
Un sentito grazie va anche agli amici incontrati durante il mio percorso di studi lontano da casa, con i quali ho condiviso le gioie e i momenti difficili della vita universitaria. Sono persone speciali con le quali si sono creati rapporti incredibili e ricordi preziosi che porterò per sempre con me.

Stockholm, April 2021
Carlo Loris Palombo