UPC CTTC

Efficient Wall Modeling for Large Eddy Simulations of General Non-Equilibrium Wall-Bounded Flows.

Centre Tecnològic de Transferència de Calor
Departament de Màquines i Motors Tèrmics
Universitat Politècnica de Catalunya

Joan Calafell Sandiumenge

Doctoral Thesis

Efficient Wall Modeling for Large Eddy Simulations of General Non-Equilibrium Wall-Bounded Flows.

Joan Calafell Sandiumenge

TESI DOCTORAL

presentada al

Departament de Màquines i Motors Tèrmics
E.S.E.I.A.A.T.
Universitat Politècnica de Catalunya

per a l'obtenció del grau de

Doctor per la Universitat Politècnica de Catalunya

Terrassa, Maig 2019

Efficient Wall Modeling for Large Eddy Simulations of General Non-Equilibrium Wall-Bounded Flows.

Joan Calafell Sandiumenge

Directors de la Tesi

Dr. Assensi Oliva Llena
Dr. Francesc Xavier Trias Miquel

Tribunal Qualificador

Dr. Roel W.C.P. Verstappen
University of Groningen

Dr. Antonio M. Pascau Benito
Universidad de Zaragoza

Dr. Joaquim Rigola Serrano
Universitat Politècnica de Catalunya

Aquesta tesi està dedicada a

els meus pares, Marta i Eduard,

a la meva germana Nina i al Lito,

i als meus nebots, Júlia, Eduard i Andreu,
que espero que igual que jo amb aquesta tesi,
trobin el camí que els faci feliços.

Agraïments

M'agradaria començar els meus agraïments per l'Assensi, per haver cregut en mi i haver-me donat la oportunitat d'iniciar la meva carrera d'investigador en un situació bastant fora del que és habitual. Molt probablement, si no hagués estat per ell, no només aquesta tesi no hagués estat possible, sinó que jo no hagués estat mai candidat a doctor. A més a més, el seu entusiasme m'ha ajudat a tirar endavant en moments difícils durant el procés® de la tesi, que també n'hi ha hagut!

A l'Oriol i la Ivette per a introduïr-me al món de la turbulència i la aerodinàmica computacional on finalment he trobat allò que m'ha omplert.

Al Xavi Trias, que va agafar el relleu compartint generosament amb mi el seus increïbles coneixements sobre turbulència, mètodes numèrics i tants altres camps, i que encara avui en dia, em segueix meravellant! També agaïr-li el fet d'ensenyar-me a ser rigorós en el treball, cosa indispensable en un científic.

Al Professor Jaume Peraire per haver-me donat la gran oportunitat de fer una estada al departament d'Aeronautics and Astronautics de l'MIT, i poder haver viscut la experiència del que és fer recerca als Estats Units.

Al Jordi Chiva. Si a algú li sembla que aquesta tesi ha durat massa, només dir que sense ell encara hauria durat més! Gràcies per la teva paciència i per haver-me ajudat a resoldre tots, i subrratllo TOTS, els centenars de problemes que et dec haver vingut a comentar durant aquests anys.

A l'Octavi i al Ramiro, que sense la seva inestimable ajuda, la meitat dels casos presentats en aquesta tesi mai haguessin vist la llum.

Al Guillem, per haver-me ajudat tantes vegades amb el LaTeX. Sense anar més lluny, per poder posar la dièresi a la *i* en el títol d'aquesta secció, que encara que pugui semblar trivial, no ho és!

I a la beca *Talent Empresa* de l'Agència d'Ajuts Universitaris i de Recerca (AGAUR) per haver-me finançat durant part d'aquest periode.

Si alguna cosa he guanyat durant aquest temps, tant o més important que els coneixements científics, han estat amistats i experiències que m'han fet viure una de les millors etapes de la meva vida:

Gràcies a tots els amics de la *guarderia* que em vau acollir de la millor manera possible i amb els quals segueixo i seguiré compartint coses, Lluís Jofre, Joan Farnós, Chamo, Aleix Báez, Joan López, Àlex Sadurní i Nico Ablanque.

Als amics del *307*, la Roser, el Xavi Trias altre cop!, i al nostre estimat inflitrat Arnau! Gràcies a tots tres per les bones estones que hem passat al nostre cau i pel suport personal en moments durs.

Als amics del gimnàs que han fet que la màquina el.líptica fós una mica menys pesada i avorrida, l'Octavi, en Chamo, en Xavi, en Nico, l'Aleix, la Nina i en Xian.

Als companys del menjador, que han fet de les tertúlies del dinar, un dels moments més esperats del dia! En Ramiro, en Xavi Álvarez, en Deniz, en Carles Oliet, i molts d'altres que ja han anat apareixent més amunt!

I a la resta de companys del laboratori, tant veterans que ja han marxat, com nouvinguts, amb tots els quals he compartit bones experiències, l'Eduard Bartrons i el seu entusiasme, la Leslye, en Víctor, en Dani Martínez, en Willy, en Ricard, en Jesús Ruano, en Kike, en Pedro Galione, en Federico, en Jesús Castro, en Quim Rigola, en David Pérez-Segarra, en Jordi Muela, l'Hector, l'Imanol, l'Ángel, l'Àdel, l'Oscar, en Firas, en Nacho, l'Ali, l'Eugenio, l'Olga, en Néstor, en Sergio, en Giorgos, en Manolo i en Mohamed.

Als amics de l'MIT que van fer de la meva estada a Boston una experiència inolvidable, en Pablo, l'Alexandra, en Sébastien i en Ferran.

De fora del laboratori també voldria agraïr l'amistat i els consells d'en Jordi Júlvez, que si no hagués estat pel seu exemple com a investigador, jo mai hauria decidit fer el doctorat. I també a en Darko, per haver-me acollit a Boston i haver-me fet sentir acompanyat des del primer moment en que vaig arribar.

Evidentment, a tot els amics de fora de la feina que fan que la meva vida sigui feliç.

I finalment, a la meva família, que sense ells, absolutament res de tot això hagués estat possible. A la meva mare Marta, al meu pare Eduard, a la meva germana Nina i el seu gran company de vida, en Lito, i als ninus, la Júlia, l'Edu i l'Andreu.

Abstract

The main purpose of this thesis has been to contribute to the development of methodologies for wall modeling Large Eddy Simulations (LES) of wall-bounded flows, especially those at high Reynolds numbers. This flow configuration is widely found in a vast range of industrial applications. Unfortunately, given the physical nature of boundary layers, their accurate numerical resolution can be computationally unaffordable. Boundary layers are multi-scale flow structures whose energetic and dynamically important motions are located in its innermost region. This layer becomes thinner and thinner as the Reynolds number is increased. Given that large flow structures strongly depend on these small motions, it is difficult to obtain accurate results if they are not properly resolved. Wall modeling aims at reproducing the inner layer effects on the flow without resolving it explicitly. This allows performing accurate numerical simulations of high Reynolds number wall-bounded flows at a fraction of the cost that would be incurred if the inner layer was temporally and spatially resolved.

This work comprises six chapters. The first one is an introduction to the existing Computational Fluid Dynamics (CFD) approaches, from the most accurate and general methodologies to the most simplified and specific techniques. In particular, the focus is placed on the resolution methodologies for wall-bounded flows, ranging from simple equilibrium boundary layers to the most complex conditions with strong pressure gradients and flow separations. In this context, a new set of expressions intended to estimate the gains of wall modeling compared to explicitly resolving the inner layer are given. The presented expressions, unlike in previous attempts, they take into account not only the costs associated with the mesh resolution, but they also consider temporal aspects. This provides a more realistic estimation of the total computational cost associated with a given high fidelity CFD simulation. The benefits of wall modeling are strongly highlighted when time integration costs are also considered. At the end of the chapter, a summary of all existing methodologies is given. The techniques are classified by their degree of approximation as well as their associated computational cost.

The second chapter introduces relevant physical magnitudes to be analyzed to ensure the reliability of a given high fidelity CFD simulation. Spatial and temporal aspects, both crucial for a correct and accurate resolution of a turbulent flow, are considered. Different methodologies to assess the suitability of the computational domain size, the total time integration span or the grid spatial resolution, are presented. In this regard, a new approach to evaluate the adequacy of a given mesh resolution in the far field, suited for Direct Numerical Simulations (DNS) and LES, is presented. All the evaluation tools presented in this chapter are subsequently used in chapter 3 in order to verify the reliability of the validation tests.

In the third chapter, a Two-Layer wall shear stress model (TLM) for LES and suitable for non-equilibrium flows and complex geometries is presented. Wall shear stress models in general, and RANS-based wall models (WM) in particular, are affected by the "log-layer mismatch" (LLM) and the resolved Reynolds stresses (RRS) inflow problems which undermine the quality of the WM numerical predictions. The model presented in this work features a temporal filter in the WM/LES interface which allows solving both problems at once with a single and low-computational-cost step. Until now, these two problems have been dealt with separately with different techniques, which in some cases were complex and computationally expensive. While the time-filtering approach was initially proposed by Yang *et al.* to address the LLM problem for a simple equilibrium wall function (Phys. Rev. Fluids 2 (2017) 104601), in this work, it is used for the first time to suppress the RRS inflow problem, actively enhancing the WM efficiency and generality compared to other existing methodologies.

On the other hand, a methodology intended to determine the optimal temporal filter length is proposed and validated in equilibrium and non-equilibrium conditions. This new technique is based on the velocity power spectrum which reveals the flow characteristic time-scales in the near-wall region. According to the results obtained in the validation tests, it is concluded that for RANS-based TLM methods, time-resolved frequencies higher than the energy-containing/inertial range limit must be filtered. It is concluded that when a time-resolved signal is used as boundary condition for a RANS domain, not only LES dissipative frequencies have to be suppressed, but also those of the inertial subrange. The velocity fluctuations corresponding to the inertial subrange scales, although not inherently diffusive, they make the RANS model working out of its range of applicability causing an overprediction of the modeled diffusivity.

Furthermore, the filter size influence on large-scale unsteady flow motions is assessed through a strongly unsteady non-equilibrium flow. It is found that for this kind of flows, the effects of the largest motions have to be taken into account since they play an essential role in global behavior. However, it remains unclear whether the value of the filter length directly influences the largest time-scales, or this is an indirect consequence of having a poorly resolved mean flow due to the use of inadequate filtering periods.

Finally, the interaction of the WM with the LES model is assessed. It is concluded that the WM acts as an integral part of the subgrid model, overriding the wall behavior of each particular LES approach.

In chapter four, the mathematical model of the TLM, based on the URANS equations, is presented. Moreover, its numerical resolution through the finite volume method is developed and finally summarized in a flow-chart. The present formulation uses an implicit time integration scheme in order to avoid the CFL restrictions on the

computational time-step size. On the other hand, the velocity-pressure coupling is solved through a fractional-step method which features a pressure-correction step with the purpose of minimizing the numerical and splitting errors of the method.

Then, in chapter five, the algorithmic implementation of the numerical model described in chapter four is presented. The TLM is a fully operational and independent CFD solver based on the URANS equations, which has been developed from scratch. The current implementation automatically generates an unstructured prismatic mesh which is partitioned according to its parallel distribution scheme, and then, the numerical model is solved within it. Additionally, the present implementation allows the bidirectional interaction between the existing LES solver and the TLM domain, both featuring completely different parallel partitions.

Given that the primary objective of wall modeling is reducing the computational costs, an efficient algorithmic and parallel implementation is a key aspect of the global modeling strategy. Thus, the parallel efficiency is evaluated through a strong scalability test. Good results are obtained although some aspects to be improved are identified.

Finally, in the last chapter, general conclusions concerning the whole work are given together with future research proposals aimed at going further in the methodologies studied in this thesis.

Contents

1

Introduction

The movement of a fluid can become a highly complex phenomenon since, under certain conditions, the fluid behavior may derive into an apparently chaotic combination of large velocity and pressure fluctuations and gradients. This highly unsteady behavior is commonly known as turbulence, and it represents a major research field in physics since this phenomenon is found in a vast range of essential industrial applications, from the wind-energy field to the automotive and aeronautical sectors.

Turbulent flows are well described from a mathematical point of view by the Navier-Stokes (N-S) equations. The equations formulation may vary depending on whether the flow is compressible or incompressible and whether the fluid's physical properties are constant or not. Nonetheless, it can be claimed that they are a fairly general mathematical and physical model which is applicable for most fluids and common flow conditions. The N-S equations for incompressible flow and constant physical properties can be written as:

$$\frac{\partial \boldsymbol{u}}{\partial t} + (\boldsymbol{u} \cdot \nabla)\boldsymbol{u} = \nabla \cdot \tau(\boldsymbol{u}) - \nabla p, \tag{1.1}$$

$$\nabla \cdot \boldsymbol{u} = 0, \tag{1.2}$$

where $\tau(\boldsymbol{u})$, is the stress tensor which is defined as $\tau(\boldsymbol{u}) = 2\nu S(\boldsymbol{u})$, being $S(\boldsymbol{u})$ the rate-of-strain tensor defined as $S(\boldsymbol{u}) = \frac{1}{2}\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T\right)$, ν is the kinematic viscosity, p is the kinematic pressure, and $\boldsymbol{u}$, is the fluid velocity field.

Turbulent flows can be seen as a cascade of kinetic energy in which the largest flow structures are continuously fed with an external supply of energy. These structures are not in equilibrium since they cannot dissipate the incoming energy at the same rate they receive it. As a consequence they split into smaller scales, transferring their energy excess to the new structures. This process is subsequently repeated until the smallest scales created during the process are capable to dissipate the energy overflow. While the energy dissipation is taken into account through the linear diffusive term $(\nabla \cdot \tau)$, the non-linear convective term $((\boldsymbol{u} \cdot \nabla)\,\boldsymbol{u})$ drives the structure

break-up process in which new motion scales are created [1]. As the weight of the convective term (inertial) rises with respect to the diffusive (dissipative) one, so does the turbulence intensity, as the convective term creates smaller and smaller structures.

In that sense, the dimensionless form of the N-S equations provides a good indicator of the turbulence level of a given flow. If we use a reference length, L_{ref}, a reference velocity, U_{ref}, a characteristic pressure, ρU_{ref}^2, and a characteristic time, L_{ref}/U_{ref}, to non-dimensionalize the N-S equations, we obtain the following expression:

$$\frac{\partial \boldsymbol{u}^*}{\partial t^*} + (\boldsymbol{u}^* \cdot \nabla^*)\boldsymbol{u}^* = \frac{1}{Re}\nabla^* \cdot [2S(\boldsymbol{u}^*)] - \nabla^* p^*, \tag{1.3}$$

where the superscript (*) denotes the dimensionless variables and operators. The new key parameter is the Reynolds number, which is defined as $Re = U_{ref}L_{ref}/\nu$. This non-dimensional magnitude measures the ratio of the inertial forces $(U_{ref}L_{ref})$ with respect to the viscous ones (ν). By analyzing equation 1.3, it can be observed that the higher the Reynolds number, the lower the diffusive term weight is. As a result, the weight of the convective term increases with the consequent creation of new smaller turbulent structures.

As previously commented, the N-S equations are an excellent mathematical and physical model for a vast range of flows. Unfortunately, in general, there are no analytical solutions for the N-S equations, and therefore, a numerical methodology has to be applied to obtain some form of solutions for practical applications. Turbulence is a time and space-dependent phenomenon as reflected in the equations (1.1). Thus, suitable temporal and spatial discretization methodologies must be considered for the numerical resolution.

Regarding the spatial discretization, there are different numerical techniques to carry it out, finite differences, finite volumes, and finite elements. All of them are numerical approaches with a common purpose; obtaining a discrete solution of the N-S equations in a finite set of points distributed throughout the computational domain. Nonetheless, the higher the Reynolds number and the smaller the flow structures, the larger the number of discrete points necessary for a correct flow resolution. This poses a major difficulty for the numerical resolution of the N-S equations since it is absolutely necessary to represent the smallest flow structures. These flow scales play the essential role of dissipating the kinetic energy overflow, keeping the balance between the energy input and output. The discrete points, also known as computational nodes, can be seen as the pixels in a digital picture. The smaller the details in a picture, the higher the necessary resolution to capture them correctly.

On the other hand, the temporal evolution has to be also integrated. There are different techniques to perform the discretization which can be mainly classified in explicit and implicit methods. Nonetheless, in both approaches, the advancement in

time is performed through discrete steps separated by a finite fraction of time. The size of the time fraction, mostly known as time-step, Δt, is also of capital importance regarding the proper resolution of the flow physics. The discrete temporal steps can be seen as the frames in a video while the time-step size is related to the number of frames per second. Any event occurring between two frames will not appear in the video, and therefore, the smaller the temporal spacing between frames, the better are represented the real events captured in the scene. Analogously, any flow structure with a characteristic time-scale smaller than the time-step will be missed from the simulation, and therefore, its physical effects as well.

1.1 Direct Numerical Simulation

It can be claimed that theoretically, a general numerical methodology could be defined to solve the N-S equations. This general methodology should be able, among other important considerations, to represent the whole energy spectrum -from the largest to the smallest scales- from a spatial and temporal point of view. This methodology actually exists, and it is called Direct Numerical Simulation (DNS). Unfortunately, in practice, its application is unfeasible in most cases given the unaffordable computational resources required to take into account all the flow scales [2]. Therefore, other numerical techniques must be introduced to circumvent this major limitation. The strategy is based on emulating the effects of the small scales through physical and mathematical models, instead of explicitly revolving them. However, these approaches may entail a significant loss of accuracy and generality of the resolution methodologies depending on the physical assumptions applied to the model formulation.

Hereafter, the most important and common approaches will be described and ordered from most to least general models.

1.2 Large Eddy Simulation

Large Eddy Simulation (LES) is a turbulence modeling approach with a low degree of physical assumptions. This makes LES a rather general and accurate methodology from both, the spatial and temporal viewpoints. This method is based on resolving the spatially-filtered N-S equations which only take into account the large structures of the flow. Conceptually, the non-resolved (sub-filter) smallest scale effects are taken into account by adding an extra viscosity to the diffusive term, making the smallest resolved scales to artificially dissipate the kinetic energy that would have had to be dissipated by the non-resolved scales.

The mathematical derivation of this approach is presented hereafter. Firstly, a filter with filter length Δ is applied to the N-S equations (1.1) in order to suppress all scales smaller than Δ. Assuming that the filtering operator $(\bar{\cdot})$ commutes with differentiation, the following expression is obtained:

$$\frac{\partial \overline{\mathbf{u}}}{\partial t} + (\overline{\mathbf{u}} \cdot \nabla)\overline{\mathbf{u}} - \nabla \cdot \tau(\overline{\mathbf{u}}) + \nabla \overline{p} = \nabla \cdot (\overline{\mathbf{u}}\,\overline{\mathbf{u}}^T - \overline{\boldsymbol{u}\boldsymbol{u}^T}), \tag{1.4}$$

where the right-hand side of the equation represents the effects of the non-resolved scales on the large flow structures with velocity $\overline{\mathbf{u}}$. Therefore, it makes sense that this term is a function of the unfiltered velocity field, $\boldsymbol{u}$, given that it is the only one containing information about the smallest filtered scales. This dependence has to be removed since, in practice, the non-filtered velocity filed is unknown. An additional assumption is then introduced in order to model the non-resolved effects as a function of the filtered velocity field. The term $\overline{\mathbf{u}}\,\overline{\mathbf{u}}^T - \overline{\boldsymbol{u}\boldsymbol{u}^T}$ is approximated as the subgrid stress tensor τ_{sgs}:

$$\tau_{sgs} \approx -\left(\overline{\mathbf{u}}\,\overline{\mathbf{u}}^T - \overline{\boldsymbol{u}\boldsymbol{u}^T}\right), \tag{1.5}$$

Then, τ_{sgs} is evaluated through the Boussinesq hypothesis, which is defined as:

$$\tau_{sgs} = -2\nu_{sgs} S(\overline{\mathbf{u}}) + \frac{1}{3} tr(\tau_{sgs}) I, \tag{1.6}$$

where I is the identity matrix, and ν_{sgs}, is the subgrid viscosity. Replacing equations 1.5 and 1.6 into 1.4, we obtain:

$$\frac{\partial \mathbf{v}}{\partial t} + (\mathbf{v} \cdot \nabla)\mathbf{v} = \nabla \cdot \left[2(\nu + \nu_{sgs}) S(\mathbf{v})\right] - \nabla \tilde{p}, \tag{1.7}$$

where $\mathbf{v}$ is a new variable which is an approximation of $\overline{\mathbf{u}}$ derived from expression (1.5), and $\tilde{p}$ is a modified pressure which includes the hydrostatic part of the Boussinesq relation. It is important to notice that Equation 1.7 is the same as the original N-S equations (1.1) except for the diffusive term, which includes an additional viscosity (ν_{sgs}) representing the damping effects of the smallest non-resolved scales.

A closure LES model is then needed to evaluate ν_{sgs}. A wide variety of different formulations can be found in the literature. Their accuracy and range of applicability strongly depend on their physical and mathematical formulation generality and the degree of realism of the applied assumptions. A detailed review of the different existing LES models was published by Zhiyin [3] in 2014. On the other hand, some of the most relevant LES approaches and their properties will be commented in more in-depth detail in Chapter 3.

Unfortunately, the LES approach has still an excessive computational cost[4] for most relevant industrial flows due to the large amount of spatial and temporal resolution required, hampering the spread of its use throughout the industry.

1.3 Reynolds-Averaged Navier-Stokes equations

The Reynolds-Averaged Navier-Stokes equations (RANS) is a widely used approach due to its significantly lower computational cost compared to the DNS and LES methodologies. The RANS strategy aims at obtaining the time-averaged velocity field instead of obtaining an accurate time-resolved flow. Given the time-scale size of RANS solutions, most of the flow structures are not represented, entailing significant savings in both, time and spatial resolutions. However, these savings are at expenses of the model application generality (given the large number of strong physical assumptions) as well as of the results quality. Given the reduced amount of time-resolved information contained in the solution, RANS models fail in predicting second-order statistics such as root-mean-square (rms) velocity fluctuations. However, first-order statistics like mean velocity and pressure fields may be obtained with an acceptable accuracy level for certain complex flows if the model is adequately tuned. Usually, in these cases, the simulation set up requires significant expertise in CFD techniques and previous in-depth knowledge of the simulated flow, making the use of RANS far from being straightforward. Despite these major drawbacks, this methodology is still attractive for specific industries since practical industrial design data may be obtained with a reasonable computational cost.

From a mathematical point of view, in the RANS approach, the instantaneous velocity field is decomposed into a mean component and a fluctuating term:

$$\boldsymbol{u} = \langle \boldsymbol{u} \rangle + \boldsymbol{u}', \tag{1.8}$$

$$\langle \boldsymbol{u}' \rangle = 0, \tag{1.9}$$

Replacing equation 1.8 into the N-S equations (1.1), and taking into account that the time-average of the velocity fluctuation of any amplitude is zero, we obtain a new equation for the mean flow:

$$(\langle \mathbf{u} \rangle \cdot \nabla)\langle \mathbf{u} \rangle = \nabla \cdot [2\nu S(\langle \mathbf{u} \rangle)] - \nabla \cdot \langle {\tau'}_{ij} \rangle - \nabla \langle p \rangle, \tag{1.10}$$

It has to be noticed that the $\langle \boldsymbol{u}' \rangle = 0$ condition implies that the time derivative term is zero. However, in some cases, it is necessary taking into account some degree of unsteady behavior. Therefore, the transient term can be included in Equation 1.10, resulting in the Unstead Reynolds-Averaged Navier-Stokes equations (URANS).

$$\frac{\partial \langle \boldsymbol{u} \rangle}{\partial t} + (\langle \mathbf{u} \rangle \cdot \nabla)\langle \mathbf{u} \rangle = \nabla \cdot [2\nu S(\langle \mathbf{u} \rangle)] - \nabla \cdot \langle {\tau'}_{ij} \rangle - \nabla \langle p \rangle, \tag{1.11}$$

For instance, the use of the URANS equations can be necessary when dealing with time-dependent boundary conditions. Nonetheless, the characteristic time-scales of

the boundary signal must be much larger than those of the turbulent motions. Additionally, even having stationary boundary conditions and under certain conditions, an unsteady solution might be obtained and the biggest motions of the flow can be modeled. Nonetheless, there is no control over the balance between the resolved and the modeled scales [5].

The URANS equations are of the same form as the N-S equations but with an additional term, the divergence of the so-called Reynolds stress tensor, which contains the information and emulates the effects of the fluctuating velocity field ($\langle \tau'_{ij} \rangle = \langle u'_i u'_j \rangle$). Again, in practice, the $\mathbf{u}'$ field is unknown, and therefore, the equation dependency on this term must be removed. The Boussinesq eddy-viscosity assumption for incompressible flows, which relates the Reynolds stress tensor with the mean flow velocity field is used for this purpose:

$$\langle \tau'_{ij} \rangle = 2\nu_T S(\langle \mathbf{u} \rangle) - \frac{2}{3} tr(\tau'_{ij}) I, \tag{1.12}$$

where ν_T is the turbulent or eddy-viscosity. Then, replacing equation 1.12 into equation 1.10, we obtain the following expression:

$$\frac{\partial \langle \mathbf{u} \rangle}{\partial t} + (\langle \mathbf{u} \rangle \cdot \nabla) \langle \mathbf{u} \rangle = \nabla \cdot \left[2 \left(\nu + \nu_T \right) S(\langle \mathbf{u} \rangle) \right] - \nabla \langle \tilde{p} \rangle, \tag{1.13}$$

It is important to notice that, like in the LES equations (1.7), the equations for the mean velocity field $\langle \mathbf{u} \rangle$ are of the same form as the N-S ones (1.1) but including an additional viscosity term that in the RANS approach, is called turbulent or eddy-viscosity. This extra viscosity takes into account the damping effects of the non-resolved fluctuating velocity field. Although the LES and the URANS equations apparently are almost identical, it is worth to remark that they are entirely different from a conceptual point of view. In the LES approach, the resolved filtered velocity field, $\overline{\mathbf{u}}$, contains a large portion of the real flow field information, while ν_{sgs} contains information only about of the smallest non-resolved scales. By contrast, in the RANS approach, most of the flow physics are modeled through ν_T instead of being explicitly resolved, and only the very large structures of the flow are represented in the numerical solution, $\langle \mathbf{u} \rangle$.

Like in the LES approach, in RANS, an external model has to be used to determine ν_T. Nevertheless, the RANS models are obtained with a large number of physical assumptions given the large gap between the real physics and the resolved ones. This high degree of modeling severely restricts the generality of the models and they have to be carefully selected and set up for each specific case. A complete review on existing RANS strategies with their properties and ranges of applicability was published by Alfonsi in 2009 [6].

1.4 Wall-bounded flows

Wall-bounded flows modeling is of great interest since they are present in a wide range of fundamental industrial areas ranging from the wind energy industry to the aerospace or automotive sectors among others.

1.4.1 Wall-resolved flows

In previous paragraphs, different methodologies to obtain numerical solutions for the N-S equations with different levels of accuracy have been discussed. However, the focus was placed on the resolution of the far field region, and no mention about the flow boundaries was made. The presence of solid boundaries adds a great difficulty to the numerical resolution of a turbulent flow due to the complexity of the flow structure in the wall vicinity, especially at high Reynolds numbers. This specific flow behavior near the wall region is known as a boundary layer.

A turbulent boundary layer encloses a vast range of different motion scales and steep velocity gradients. Since the behavior of the small inner layer structures has substantial implications on the development of the outer layer and by extension on the far field region, a sufficiently fine mesh is required in the near-wall area to capture them. This physical phenomenon is extremely challenging since the structures located in the inner layer become smaller as the Reynolds number increases, entailing massive requirements in terms of spatial and temporal resolution.

Although boundary layers may have different shapes and particular features depending on the Reynolds number and the problem geometry, it can be claimed that their underlying physics are rather general. To illustrate the boundary layer phenomenon, a turbulent channel flow will be used to describe the different behaviors and flow patterns within the wall layer, since the characteristic regions of this specific case can be found in many different conditions. The channel flow is a canonical case widely studied both, experimentally and numerically. The flow configuration consists of two infinite and parallel flat plates separated by a distance of 2δ. A fluid moves between them with a space-averaged velocity (bulk velocity) U parallel to the plates. This case is broadly used to assess new modeling approaches, especially those intended for wall-bounded flows, given its geometrical simplicity and the vast amount of data available for reference.

First of all, the skin friction velocity, u_τ, will be defined since different flow quantities, such as the streamwise velocity and the thickness of the different flow layers, scale with this magnitude in the near-wall region. On the other hand, in the channel core (far from the walls), the flow magnitudes scale with the bulk velocity, U, and the channel half height, δ. These quantities are useful to define in the most general possible way the different regions enclosed within a boundary layer by using non-dimensional values. The skin friction velocity is defined as $u_\tau = \sqrt{\tau_w/\rho}$,

where ρ is the fluid density, and τ_w, is the wall shear stress, which is proportional to the wall-normal velocity gradient, $\tau_w = \rho\nu \left(\partial u/\partial y\right)_{y=0}$. The dimensionless values obtained with u_τ are said to be expressed in wall units, and they are denoted with a $(+)$ superscript. Specifically, the most representative magnitudes are the streamwise velocity, $u^+ = u/u_\tau$, and the dimensionless wall distance, $y^+ = yu_\tau/\nu$.

In Figure 1.1, the flow motion scales are depicted at different wall distances to have insight into the flow behavior and structure at each different region which composes the boundary layer. The visualization is carried out by visualizing iso-surfaces of the second invariant of the velocity gradient tensor, Q. This method allows the visualization of the different flow structures, from the mostly shear-dominant (laminar) to the vorticity-driven ones (turbulent), depending on the value of Q (see Chapter 2 for more details). It can be observed that the near-wall region behavior $(y^+ < 5)$ is almost laminar. In this area, the streamwise velocity profile is linear with respect to y^+, and therefore, the wall shear stress can be easily determined without error. As the wall distance increases, the turbulent activity rises. This can be inferred from the dimensionless wall distance definition, y^+, itself. This magnitude can be seen as a local Reynolds number, $y^+ = u_\tau y/\nu$, based on the skin friction velocity (See definition of Reynolds number at the beginning of this section) and the dimensional wall distance. According to this expression, a high value of y^+ means that the inertial forces dominate over the viscous ones, triggering the turbulent activity with rising values of y^+.

In equilibrium conditions, the boundary layer behavior is universal and it is called the law of the wall [2, 7]. These conditions apply for wall-parallel turbulent flows in the absence of momentum sources and geometrical discontinuities, such as flat-plate turbulent boundary layers, channel and pipe flows. The existence of a near-wall region whose behavior is independent of the far field properties was firstly suggested by Prandtl in 1925[8]. He postulated that for the region located between the wall and $y/\delta << 1$, being δ the boundary layer thickness, there is an inner layer whose mean velocity profile in the streamwise direction is determined by the viscous scales, u_τ, and $\delta_v = \nu/u_\tau$, and it is independent of δ and the far-field velocity U. Prandtl showed that the mean velocity profile is only a function of the wall distance when expressed in terms of wall units:

$$\langle u \rangle / u_\tau = f(y/\delta_v) \longrightarrow \langle u \rangle^+ = f(y^+), \tag{1.14}$$

The $f(y^+)$ function definition varies along the wall-normal direction as the inertial forces prevail over the viscous ones, causing a transition from a laminar to a turbulent flow pattern. Depending on the flow regime, different regions within the inner layer can be differentiated. In Table 1.1, the different regions that compose the inner layer are summarized and classified according to its distance from the wall and flow regime. The table is split into a couple of subtables. Firstly, a set of layers whose properties

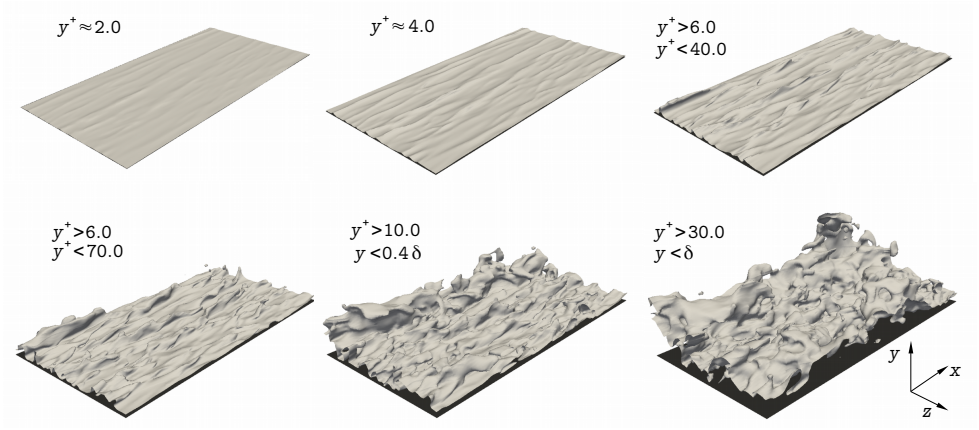

Figure 1.1: Iso-surfaces of the second invariant of the velocity gradient tensor, Q, at different wall distances of a turbulent Channel flow. See Table 1.1 to identify the different layer names and characteristics.

scale with viscous magnitudes, u_τ, is shown. In the second subtable, a more general classification is used. The first group in the second subtable, the inner layer, includes all the layers whose magnitudes scale with u_τ (all those of the first subtable), while in the second group, the outer layer, the flow parameters are almost unaffected by the viscosity effects, and they scale with outer parameters, i.e., U and δ.

The wall closest layer is the viscous sublayer, which features a laminar flow regime given that the inertial forces are too small to overcome the viscous ones. In this area, the streamwise velocity varies linearly with the wall distance. As moving away from the wall, the inertial forces rise due to the fluid acceleration, triggering flow instabilities. The buffer layer is a transition region from laminar to turbulent regime, where the flow instabilities develop until reaching a developed turbulent state in the log-law region. This layer features mixed and unstable behavior, and no universal physical pattern has been found. For that reason, the expression for $f(y^+)$ in this zone is undefined. Finally, as the fully turbulent state is reached, the mean streamwise velocity function evolves to a logarithmic profile which gives name to this specific region. The log-law region was firstly identified by Von Kármán in 1930 [9], which proposed the following expression for the mean velocity profile:

$$\langle u \rangle^+ = (1/\kappa) \ln y^+ + B, \tag{1.15}$$

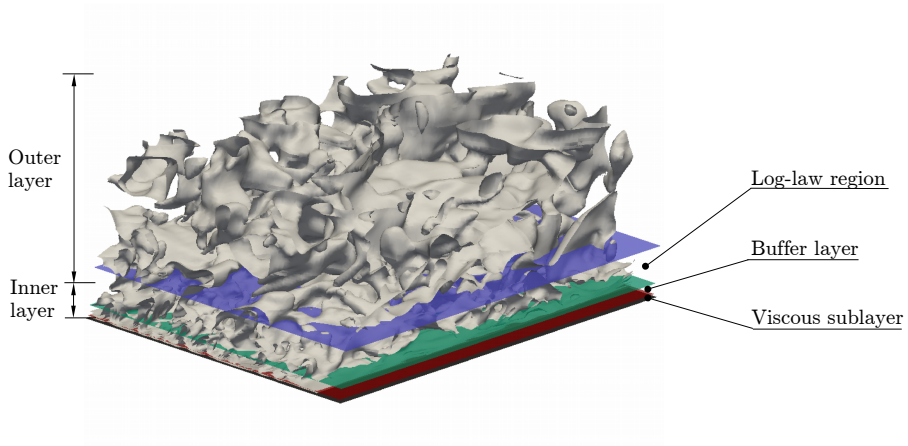

Figure 1.2: Different regions enclosed within a boundary layer of a turbulent channel flow. See Table 1.1 for details.

where κ is the Von Kármán constant whose value is approximately 0.41, and B is another constant of value approximately 5.2. It can be found different values for the log-law constants in the literature, although all of them, are within a 5% of the initial proposed values [2]. More recent works proposed alternative expressions for the mean velocity profile in the logarithmic region. In 1993, Barenblatt *et al.* showed that the velocity profile can also be expressed through Reynolds-dependent power-law solutions [10, 11]. Later, Cipra confirmed these findings through experimental evidence [12]. However, these results have not been fully accepted by other authors [13].

Table 1.1: Classification of the different regions found within a boundary layer according to their physical behavior and properties. Their location is delimited by the distance to the wall, in wall units or as a fraction of δ depending on their physical properties. [2]

Region	Location	Properties
Viscous sublayer	$y^+ < 5$	The mean streamwise velocity is linear with respect to the wall distance, both in wall units $u^+ = y^+$. Viscous forces prevail over inertial ones.
Buffer layer	$5 < y^+ < 30$	Region between the viscous sublayer and the log-law region. Turbulent instabilities are triggered
Log-law region	$y^+ > 30$ $y/\delta < 0.3$	The mean streamwise velocity has a logarithmic behavior with respect to the wall distance, $u^+ = (1/\kappa)\ln y^+ + B$. The flow regime is predominantly turbulent.
Inner layer	$y/\delta < 0.1$	The mean streamwise velocity scales with u_τ and is not affected by the far field velocity characteristic.
Outer layer	$y^+ > 50$	The effects of the viscosity on the streamwise velocity are negligible.
Overlap region	$y^+ > 50$ $y/\delta < 0.1$	Inner and outer layers are overlapped in this region (at high Reynolds numbers)

In Figure 1.3, the law of the wall, $\langle u \rangle^+ = f(y^+)$, obtained through DNS, is compared for three different equilibrium flows, a Channel flow at $Re_\tau \approx 2000$ [14], a Flat plate turbulent boundary layer at $Re_\theta \approx 6500$[15], and a Pipe flow at $Re_\tau \approx 3000$ [16]. In the plot, the different regions of $f(y^+)$ are clearly defined and it can be seen how the numerical DNS results precisely match the analytical expressions proposed by Prandtl and Von Kármán.

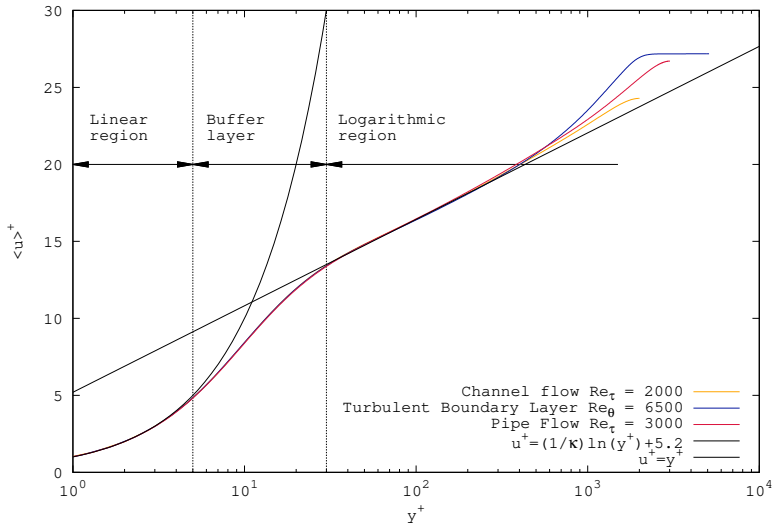

Figure 1.3: The law of the wall obtained through DNS of three different canonic flows: Channel flow at $Re_\tau \approx 2000$ [14] (orange line), a Flat plate turbulent boundary layer at $Re_\theta \approx 6500$[15] (blue), and a Pipe flow at $Re_\tau \approx 3000$ [16] (red). The expressions of Prandtl and Von Kármán are also shown.

It is worth to point out that the velocity profiles in viscous units for the three different cases, regardless of the Reynolds number and the geometrical configuration, are completely overlapped except for the area away from the wall. In this region, the far field boundary condition significantly influences the flow behavior.

To obtain reliable numerical results from a wall-bounded flow simulation, it is essential to feed the computation with an accurate wall shear stress at the solid boundary, regardless of the nature of the flow governing equations. Otherwise, the flow would be unphysically accelerated or slowed down depending on whether τ_w is over or underpredicted. The incorrect prediction of the bulk flow velocity can trigger a variety of unphysical flow phenomena, even in the far field, that would undermine the numerical results.

To obtain a correct wall shear stress value, it is imperative to place at least one computational node well below the viscous sublayer limit ($y^+ \approx 1$) throughout the solid surface. As previously commented, the velocity profile is linear in this area, and the velocity gradient can be evaluated with a negligible error. This poses a great difficulty for numerical simulations since the viscous sublayer may become

dramatically thin at high Reynolds numbers. This forces the use of extremely fine meshes with tiny computational cells in the near-wall area, prohibitively increasing the total number of nodes. A collateral consequence of the small size of the grid cells is the steep reduction of the allowed computational time step-size. The Courant-Friedrichs-Levy (CFL) number sets an upper limit for the time-step, and it is a function of the computational grid size among other parameters. The smaller the smallest grid cell, the lower the Δt limit, affecting the whole computational domain. If the used Δt is higher than the CFL constraint, unphysical results could be obtained. Additionally, the numerical stability of the simulation may be compromised in certain numerical schemes such as the explicit ones.

As for free-flows, there are several approaches in order to obtain numerical solutions for wall-bounded flows. The DNS and the wall-resolved LES (WRLES) techniques explicitly resolve the boundary layer from the spatial and the temporal point of view. They use a sufficiently large grid resolution and small time steps to adequately capture all the flow structures contained in the wall layer while modeling the far field according to the strategies presented in previous sections. However, using the DNS and LES techniques for solving wall-bounded flows is even more computationally unaffordable than for free flows given the difficulties explained above. RANS models are also suitable for modeling wall-bounded flows. Nonetheless, the grid requirements in the wall-normal directions are the same as those for DNS and WRLES, while suffering the lack of generality and accuracy commented in Section 1.3.

In order to make accurate simulations of wall-bounded flows feasible, several strategies have been proposed or are being developed to deal with this specific problem. The purpose of these techniques is to model the behavior of the boundary layer in one way or another instead of explicitly resolving it. The main target is to reduce the grid resolution requirements and consequently, allowing the use of larger time steps.

Hereafter, the main existing approaches will be detailed and classified from the most to the least accurate.

1.4.2 Wall Modeled Large Eddy Simulation

Wall modeled Large Eddy Simulation (WMLES) is a major research field given the great benefits of this approach. The strategy is based on taking advantage of the temporal and spatial accuracy of the LES methodology while avoiding the unaffordable requirements imposed by the presence of a solid wall.

Some works intended to quantify the gains of using WMLES instead of WRLES from a grid resolution point of view can be found in the literature. Chapman [17] published a first approximation in 1979 concluding that, for a WRLES (i.e., inner layer resolution) the total number of grid points, N, is proportional to $Re_{L_x}^{9/5}$. By contrast,

for a WMLES (i.e., outer layer resolution only) the mesh resolution is proportional to $Re_{L_x}^{2/5}$. These estimates were based on a turbulent boundary layer over a flat plate case, being L_x the plate length in the streamwise direction. Choi and Moin[18] updated the Chapman estimates by using more recent studies on the skin friction coefficient and boundary layer thickness. They concluded that the WRLES requires a number of grid points proportional to $Re_{L_x}^{13/7}$, while the WMLES only required a number proportional to Re_{L_x}. On the other hand, in the studies conducted by Chapman, only a slight mention to the Reynolds number scaling of the time-step was made with no further development, whereas in the works carried out by Choi and Moin, the time-step restriction was not taken into account. Nonetheless, this issue is also of capital importance, and probably even more challenging than the number of grid points since parallelization of temporal evolution, although it is a current research topic[19], is of great difficulty.

In this regard, a set of expressions approximating the Reynolds number scaling of the upper bound of the computational time-step size, Δt, is proposed hereafter. It is straightforward to show that the computational cell volume, v_{cell}, is inversely proportional to the number of grid points, N_V, in a given volume, V, when the same assumptions of Chapman and Choi and Moin apply. This volume V may be either a portion of the outer region of a boundary layer or a sample of the inner viscous layer. This can be expressed as $v_{cell} \sim 1/N_V$, and using the scalings proposed by Choi and Moin in terms of grid points, we find that:

$$v_{cell}^{WM} \sim Re_{L_x}^{-1}, \tag{1.16}$$

$$v_{cell}^{WR} \sim Re_{L_x}^{-13/7}, \tag{1.17}$$

The CFL requirements set two different upper bounds for the time-step, one for the convective, Δt_c, and another for the diffusive term, Δt_d, being used the most restrictive at each computational step. It should be pointed out that the CFL time-step is small compared to the smallest physical time-scales supported by a given mesh, which ensures that the accuracy is limited by the spatial resolution rather than the temporal one. This applies for flows in which eddies deform at a rate significantly smaller than the maximum velocity in the computational domain which holds for a large number of turbulent flows [20]. For structured meshes, the CFL bounds are expressed as:

$$\Delta t_c = C_{conv} \left| \frac{\Delta x_i}{u_i} \right|_{min}, \tag{1.18}$$

$$\Delta t_d = C_{diff} \left| \frac{\Delta x_i^2}{\nu} \right|_{min}, \tag{1.19}$$

where C_{conv} and C_{diff} are two constants which depend on the time integration scheme, ν is the kinematic viscosity, the i subindex stands for the spatial coordinates, Δx_i is the grid spacing, and u_i is the velocity component. For unstructured meshes, the grid spacing is approximated according to the classical Deardorff definition[21], $\Delta x_i = v_{cell}^{1/3}$, while u_i is replaced by the velocity magnitude. Combining equations (1.16)-(1.17) and (1.18)-(1.19), we find four expressions for Δt:

$$\Delta t_c^{WM} \sim Re_{Lx}^{-1/3} \quad , \quad \Delta t_d^{WM} \sim Re_{Lx}^{-2/3}, \tag{1.20}$$

$$\Delta t_c^{WR} \sim Re_{Lx}^{-13/21} \quad , \quad \Delta t_d^{WR} \sim Re_{Lx}^{-26/21}, \tag{1.21}$$

where Δt_c^{WM} and Δt_d^{WM} are the upper bound of the time-step for the convective and the diffusive terms in WMLES while Δt_c^{WR} and Δt_d^{WR} are the same values but for WRLES. The scaling laws vary strongly depending on which restriction prevails, the convective or the diffusive one. Nevertheless, for a given Re_{Lx}, the more restrictive diffusive bound gets dominant over the convective one as the mesh is refined [22]. Hence, in WMLES, the time-step upper bound scales closer to $Re_{Lx}^{-1/3}$ while in WRLES, the Δt scales nearly as $Re_{Lx}^{-26/21}$.

Nonetheless, the feasibility of a particular computation depends on the combined costs related to the number of grid points and the time-step size. We define the non-dimensional variable T_{cc} which stands for the total computational cost of a given computation. We are not interested in quantifying its value but to determine its scaling with the Reynolds number. Considering an ideal code, the computational cost would be proportional to the number of grid points ($T_{cc} \sim N$), while if a direct solver is considered for the Poisson equation, the computational cost per time-step would be also inversely proportional to the time-step size ($T_{cc} \sim 1/\Delta t$). Combining both scaling laws and replacing the N and Δt values for the WMLES and WRLES Reynolds scaling expressions respectively, we obtain:

$$T_{cc}^{WM} \sim N\frac{1}{\Delta t} = Re_{Lx}\frac{1}{Re_{Lx}^{-1/3}} = Re_{Lx}^{4/3}, \tag{1.22}$$

$$T_{cc}^{WR} \sim N\frac{1}{\Delta t} = Re_{Lx}^{13/7}\frac{1}{Re_{Lx}^{-26/21}} = Re_{Lx}^{65/21}, \tag{1.23}$$

where T_{cc}^{WM} and T_{cc}^{WR} are the total computational cost in a WMLES and in a WRLES computation, respectively. While the total computational cost scaling of a WMLES suffers a slight change when considering the time-integration contribution compared to the Choi and Moin estimates (from Re_{Lx} to $Re_{Lx}^{1.33}$), in a WRLES the T_{cc} scaling increases dramatically from approximately $Re_{Lx}^{1.85}$ to $Re_{Lx}^{3.09}$. These new expressions highlight even more the extreme difficulty of performing WRLES computations of industrial relevant Reynolds numbers.

After reviewing the difficulties that arise when dealing with WRLES, it is clear that WMLES is a very promising strategy to make wall flow computations at high Reynolds number affordable. Several approaches have been developed to deal with the boundary layer complexity. All of them are based on modeling the inner layer in one way or the other instead of fully resolving it from a temporal and spatial point of view. These strategies can be mainly split into two major subgroups, hybrid RANS-LES models, and wall shear stress methods. The main conceptual difference between both methodologies is that in hybrid RANS-LES, the wall model (WM) is embedded in the turbulence model itself. The model returns a RANS-type turbulent viscosity in the near-wall regions while it switches to an LES-type subgrid viscosity in the outer layer and the far field regions. On the other hand, in wall shear stress methods, an LES model is solved down to the wall while a shear stress is provided by an external model to the LES domain as a boundary condition. The main differences and the range of applicability of these methodologies were discussed in detail by Piomelli[4] in 2008, while a complete review of the hybrid RANS-LES techniques and their continuous improvements was published by Spalart et al.[23] in 2009.

Wall Shear Stress Methods

This strategy is based on solving the whole computational domain with an LES model, but with an strongly under-resolved wall region. In these conditions, the wall shear stress would be clearly mispredicted given that, at high Reynolds number flows, the first off-wall nodes could be easily placed over $y^+ > 100$, far away from the viscous sublayer. In order to avoid the undesirable effects of a flawed τ_w evaluation, an external model is used to supply an accurate wall shear stress value.

There are many different methodologies intended to properly evaluate the wall shear stress, from simple algebraic models with a high degree of modeling derived with strong assumptions to highly accurate formulations in which complex governing equations are solved. The range of applicability of the models strongly depends on its mathematical and physical formulation generality.

Wall functions

The most straightforward approach is known as wall function. This method relies on an a priori known velocity profile based on the law of the wall. The models belonging to this group are called functions given that the evaluation of the wall shear stress is performed through an algebraic expression which, in general, uses the LES solution as input data. In general, they take the following form:

$$\tau_w = g(u_1, y_1), \tag{1.24}$$

where τ_w is the wall shear stress, u_1 is the tangential velocity at the first off-wall LES node, and y_1, the node distance from the wall. The approach is based on modifying the law of the wall expression, $\langle u \rangle^+ = f(y^+)$, to obtain an expression of the form of Equation 1.24. The function $\langle u \rangle^+(y^+)$ can be the log-law, the power law, a linear viscous law or even a numerical interpolation obtained from DNS data. Nonetheless, it has to be taken into account that the law of the wall is only valid on average, and therefore, the computed wall shear stress will also be averaged in time. Moreover, given that the law of the wall is only valid for equilibrium flows, the wall functions have a somewhat limited range of applicability. By contrast, the main advantage of these models is their mathematical simplicity and extremely low computational cost given that no governing equations have to be solved.

The concept of wall function was initially introduced in the 1970s by Deardorff [21]. He used slip boundary conditions to compute a plane Poiseuille flow at very high Reynolds number on a coarse mesh. The value of the wall parallel velocity at the boundary was obtained according to a logarithmic expression derived from the time-averaged law of the wall. In order to obtain a more realistic physical behavior, a methodology to introduce temporal fluctuations on the wall shear stress signal was proposed by Schumann in 1975 [24]. He proposed to assume that the proportionality between the average values of $\langle u_1 \rangle$ and $\langle \tau_w \rangle$ also holds for the instantaneous values. Since the instantaneous values of u_1 are obtained from the LES computation, the instantaneous wall shear stress can also be obtained:

$$\tau_w = \frac{\langle \tau_w \rangle}{\langle u_1 \rangle} u_1, \tag{1.25}$$

Subsequent improvements were developed by different authors [25, 26, 27, 28], some of them including surface roughness effects in the model formulation. On the other hand, Werner and Wengle used the 1/7-power law instead of the more commonly used log-law. This allowed a direct evaluation of $g(u_1, y_1)$ [29], avoiding the use of an iterative process and making the resolution algorithm even more efficient from a computational cost point of view.

Zonal or Two-Layer wall models

In order to avoid relying on a priori known flow data, the zonal or two-layer methods (TLM) were initially proposed by Balaras et al. [30, 31]. This methodology is based on the resolution of the URANS equations or a simplified variant in a fine auxiliary mesh that is embedded in the LES domain between the solid boundary and the first off-wall node (see Figure 3.1). The boundary conditions are taken from the LES nodes while the time integration is carried out with implicit or semi-implicit schemes to avoid temporal restrictions due to stability considerations.

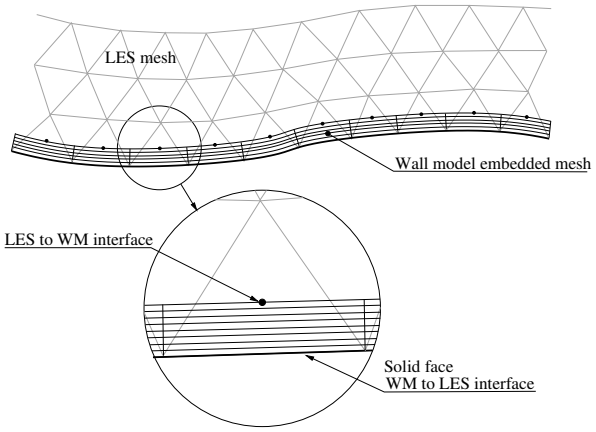

Figure 1.4: The two-layer strategy scheme.

With increasing complexity of the near-wall governing equations, fewer assumptions are made compared to other methodologies such as wall functions. Once the near-wall velocity profile is established, an accurate wall shear stress can be evaluated and fed back into the LES domain. In the initial works[30, 31], the two-dimensional turbulent boundary layer equations (TBLE) were solved. These equations are derived from the URANS ones by assuming the non-variation of the pressure in the wall-normal direction (i.e., $\partial P/\partial n = 0$), whereas only the variation of the velocity in the wall-normal direction is considered for the diffusive term. To close the formulation, a simple mixing-length eddy viscosity model with a damping function was used to evaluate the turbulent viscosity, ν_t. In the initial work, Balaras *et al.* achieved promising results in the performed numerical tests. However, only equilibrium flows were tested (channel and duct flow) and they suggested to carry out further tests with strong adverse pressure gradients and separated flow regions. From the initial concept, different authors have suggested variants and improvements, and the idea is still being developed.

Cabot *et al.* [32, 33] carried out computations on a backward-facing step. They found that the total amount of Reynolds stresses (resolved + modeled) in the wall layer was overestimated when using the TBLE-based (non-equilibrium) model. In RANS modeling, almost the total amount of Reynolds stresses is generated by the model. Nonetheless, in TLM, resolved Reynolds stresses (RRS) are carried by the advective term from the LES to the RANS domain through the boundary condition, causing an overprediction of the total Reynolds stresses and hence, also of the skin friction coefficient C_f. To counteract this effect, the authors proposed a modified

mixing-length turbulence model in which the Von Kármán constant, κ, was adjusted dynamically to reduce the modeled Reynolds stresses contribution. This correction allowed obtaining better results in the skin friction prediction. They also tested the so-called "stress balance model" that basically neglects the pressure gradient and advective terms of the TBLE equations. They found that while the mean velocity profiles were relatively well resolved compared with the WRLES, the model failed in predicting the C_f in the separated region or in the presence of strong pressure gradients. The results showed that neglecting the advective and pressure gradient terms was an unreasonable assumption when it comes to non-equilibrium flows. All these results were confirmed in subsequent works. Namely, Wang [34] performed computations of a trailing-edge flow reaching similar conclusions with both models, the equilibrium (only diffusive and temporal terms) and the full TBLE one. Especially good results were obtained for the skin friction coefficient when using the dynamic correction proposed by Cabot[32] compared to the ones obtained with a constant κ in which the C_f was overestimated. Later, Tessicini [35, 36] performed computations of flows with more complex geometries, such as a fully three-dimensional (3D) circular hill. The equilibrium model was tested together with a variant of the TBLE model which only neglected the advective term, making clear again the need for the non-equilibrium terms in separated flow regions. Finally, the same conclusion was reached by Kawai *et al.* [37] when computing the flow around an airfoil near stall condition.

The equilibrium assumption is also implicit with the use of a mixing-length eddy viscosity model. Diurno *et al.* [38] also computed a backward-facing step case but using the Spalart Allmaras (SA) model[39] in the wall layer to override the equilibrium assumption. Even though the overprediction of Reynolds stresses pointed out by Cabot[32] concerns any RANS model, they found that the WM performed slightly better in the separated regions regarding the C_f prediction when using SA instead of the simple mixing-length with constant κ. However, no evident improvements were obtained in other quantities such as the mean velocity profiles. For more details, the reader is referred to the extensive review of wall-layer modeling published by Piomelli and Balaras [40] in 2002.

From 2010 onwards, new efforts have been carried out to improve the TLM strategy. Some of the previous works pointed out the impossibility of having accurate LES velocity data at the first few off-wall nodes, mainly due to the use of inadequate LES models and large numerical errors in the near-wall region when using coarse meshes [33]. These errors were supposed to cause the so-called "log-layer mismatch" (LLM) error, an unphysical rise of the fluid velocity in the boundary layer logarithmic region. Kawai and Larsson [41, 42] stretched the wall layer mesh beyond the first off-wall node, reducing numerical and subgrid modeling errors caused by the wall proximity and suppressing the LLM issue. Simultaneously, Kawai and Larsson also proposed a new approach to evaluate the κ coefficient [37, 41, 43] in the TLM

turbulence model. They showed that κ could not remain constant in the wall-normal direction given that the ratio between resolved/modeled Reynolds stressed varies dramatically in that direction. They argued that the wall-parallel resolution was fine enough to resolve turbulent scales from a given wall separation since the size of the near-wall turbulent structures is proportional to the wall distance. As the wall-parallel grid spacing gets smaller with respect to the wall-normal distance, the contribution of resolved stresses to the total ones rises, and the modeled part must be lowered to avoid overpredictions. Both contributions, the stretched mesh and the variable κ were successful in predicting mean velocity profiles and turbulence statistics in various tests, such as a turbulent boundary layer and a compressible shock/boundary-layer interaction among others. Nevertheless, the dynamic procedure proposed by Kawai and Larsson[37, 41, 43] was dependent on a model coefficient that indicated the height at which the generation of resolved stresses began. To circumvent this, a parameter-free model was proposed by Park and Moin [44] in which the resolved part of the Reynolds stresses was determined explicitly and subtracted to the RANS modeled component. In the latest implementations of TLM[37, 43, 44], the full URANS compressible equations have been used instead of the TBLE ones which were typically used so far, in order to generalize the method and to take into account all the non-equilibrium phenomena.

Another factor to take into account was the generalization of the method for any geometry. Patil *et al.* [45] solved the TBLE equations in a generalized coordinate system that allowed the model to adapt to complex geometries. On the other hand, Bodart and Larsson [46] developed a numerical strategy to implement the methodologies proposed by Kawai and Larsson[37, 41, 43] in unstructured meshes. The strategy was followed by Park and Moin [44] who tested their model on a NACA 4412 airfoil at the angle of attack of 12°, in which the laminar to turbulent transition was taken into account by switching off the RANS ν_t in the laminar region by using a turbulent kinetic energy sensor [47]. Reasonably good results were obtained regarding mean velocity profiles and pressure coefficient C_p. A complete review of wall modeling and in particular of two-layer models was published by Larsson *et al.*[48] in 2016.

Hybrid RANS-LES Models

The Hybrid RANS-LES approach is based on modeling the whole boundary layer region through a RANS model while keeping a high accuracy LES method for the far field area.

A first successful formulation of this methodology was proposed by Spalart in 1997 [49], and was initially designed for external flows with thin boundary layers and mass-separated regions. The first implementation of this methodology is known as

Detached Eddy Simulation (DES), and was based on the resolution of the transport equation for a modified turbulent viscosity proposed by Spalart and Allmaras[39] (SA). The methodology performed reasonably well in external flows featuring a RANS-type mesh in the wall region (i.e., large grid spacing in the wall-parallel direction and fine resolution in the wall-normal direction). To switch from the RANS to LES mode, the wall distance parameter used in the SA equation was replaced by a distance based on the control volume size, acting as an LES filter length scale. This strategy implied that the RANS/LES interface was not determined by physical considerations but for geometrical ones in which the way the mesh was constructed played an important role. This approach may lead the interface to move closer to the wall if the wall-parallel grid spacing is smaller than the boundary layer thickness, causing a premature switch off of the RANS-modelled Reynolds stresses at a moment at which the mesh is not still able to balance it with resolved LES stresses. This phenomenon is known as modeled stress depletion (MSD), and it may cause "grid induced separation" [50] in boundary layers approaching separation.
In 2001, Strelets generalized the definition of DES for any RANS model and extended the initial implementation to the Menter $k-\omega$ (M-SST) formulation [51]. The motivation of the proposed methodology was avoiding the difficulty of the SA model in predicting separation accurately by replacing it with a more reliable model from the detachment prediction point of view. Additionally, the characteristic length scale was not depending on the wall distance as in SA, but it was provided by the model itself with a more physical basis. However, no clear advantages were found in using the M-SST-based model with respect to the SA one.

The Original DES was designed to model the entire attached boundary layer regions with RANS technology. However, Nikitin et al. [52] attempted to only model the inner boundary layer with RANS, using the initial DES formulation as a simple wall model for LES. They computed several channel flows at high Reynolds numbers with an LES-type mesh (i.e., similar grid refinement in all directions). A problem arose in the RANS/LES transition region called "log-layer mismatch." The velocity profiles in the LES and the RANS regions were misaligned due to a 15% underprediction in the skin friction coefficient. This caused a local inaccuracy in the velocity gradient prediction in the wall-normal direction at the transition point. The main reason for this unphysical phenomenon is an imbalance of momentum caused by the transition between two regions with different ratios of resolved/modeled Reynolds stresses. Several studies have been done in order to mitigate this drawback. A successful solution was proposed by Piomelli et al. [53] They used a stochastic backscatter model introduced in the inner layer through a forcing term in the momentum equation. The model provided an appropriated level of fluctuations to the LES region by increasing the resolved part of Reynolds stresses and corrected the unphysical mean velocity

profile. However, the method was very sensitive to the forcing amplitude, and improvements were introduced by Larsson et al. in order to control the forcing magnitude dynamically [54]. More recently, significant modifications were proposed for DES approach in order to enhance its performance. In 2006 Spalart proposed Delayed DES (DDES) [55] in which the RANS/LES transition criterion was made independent of the mesh construction through a new definition of the DES length scale that depends on the eddy-viscosity field. Later, Shur et al. [56] proposed a modification of DDES to deal with inflow turbulence. Empirical functions were also added in order to increase the resolved Reynolds stresses near the wall, helping to correct the "log layer mismatch" issue. This new version of DES was named as IDDES (Improved Delayed DES). A detailed review of the DES technique and its continuous improvements was published by Spalart et al. in 2009 [23]. The DES concept is explained and the main strengths and weaknesses of the method, together with the proposed corrections, are presented.

1.4.3 Wall functions for RANS

Although the space and temporal resolution requirements of RANS models are significantly lower than those for LES, still, the grid resolution on the wall-normal direction is also a critical issue in the RANS approach. As in LES, it is needed to place the first off-wall node in the viscous sublayer $y^+ \approx 1$ [57] to obtain a correct wall shear stress evaluation. In order to mitigate this severe constraint, wall functions may be used to estimate the wall shear stress instead of directly compute it through the computational grid. The strategy is the same as the one used for LES wall functions, although the mathematical and physical formulations are adapted for the RANS context. A study on the wall behavior of the most common RANS formulations was published by Kalitzin et al. [57]. In the same work, a new wall function suitable for any RANS model is described. The proposed formulation belongs to the adaptative wall function type which does not require to place the first off-wall nodes at any specific height with respect the solid face [58].

1.4.4 Potential flow

In the potential flow theory, a set of fluid motion governing equations is derived under strong assumptions. The flow is supposed to be irrotational, inviscid and incompressible. According to these hypotheses, the fluid velocity field can be expressed as:

$$\mathbf{u} = \nabla\phi, \tag{1.26}$$

where ϕ is the scalar potential function which fulfills the basic laws of fluid mechanics, i.e., mass and momentum conservation. On the other hand, the following vector identity holds for any scalar field ϕ:

$$\nabla \times \nabla\phi = 0, \tag{1.27}$$

Therefore, for equation 1.26 to be applicable, the flow must be irrotational:

$$\nabla \times \mathbf{u} = \nabla \times \nabla\phi = 0, \tag{1.28}$$

Taking the divergence of equation 1.26 and applying the incompressibility constraint $(\nabla \cdot \mathbf{u} = 0)$, we obtain the Laplace equation for the potential function:

$$\nabla \cdot \mathbf{u} = \nabla \cdot \nabla\phi \longrightarrow \nabla^2\phi = 0, \tag{1.29}$$

The Laplace equation is a linear PDE which has an analytical solution for certain simple boundary conditions. Nonetheless, being a linear equation, more complex solutions can be obtained as linear combinations of the simpler ones.

Although it is true that the physical assumptions are very restrictive, they apply in certain circumstances. The linear properties of the Laplace equation make the required computational cost to solve it to be almost negligible, especially if compared with the cost of resolving any variant of the N-S equations. This makes the potential flow theory a very interesting approach provided that the flow conditions fulfill the restrictive constraints.

One of the major drawbacks of the potential theory is the inviscid hypothesis, which hampers its applicability to wall-bounded flows. The inner part of a boundary layer is mostly governed by viscous effects, and they cannot be neglected to obtain a reasonable numerical solution. Nonetheless, the potential flow theory can be coupled with a viscous boundary layer model to overcome this major weakness [59]. Additionally, corrections can also be applied to the incompressible solution to perform compressible flow computations at low Mach numbers.

Several applications have been developed for aerodynamic computations of very specific geometries and flow conditions[60]. In general, the range of applicability of these codes is very narrow, but when applicable, the ratio of results quality with respect to the computational cost is unbeatable. Another limitation of this approach is the amount of flow data obtained in the simulation. In general, only information on the velocity and pressure fields is obtained. However, this data is sufficient for most practical industrial design processes.

1.5 Summary of methodologies

A summarizing table (1.2) comprising all the methodologies discussed above, is presented hereafter. It includes the range of applicability for each methodology as well as their computational cost (CC) and the strength of their physical assumptions, represented by the degree of modeling (DOM). In general, the stronger the assumptions, the lower the model generality and its ease of use. Increasing the DOM, usually entails that model coefficients and boundary conditions for some specific quantities must be provided and tuned to obtain correct flow prediction. Some of these specific quantities may range from turbulent kinetic energy to energy dissipation, and they may not be easily determined. On the other hand, although some models might be applicable for complex non-equilibrium conditions, a strong a priori flow knowledge is needed to adjust the model properly and obtaining good results. Additionally, the DOM is usually detrimental to the temporal and spatial resolutions, and therefore, much fewer details of the flow behavior can be obtained, especially for quantities related to the temporal behavior such as velocity fluctuations or large structure motion frequencies.

1.6 Conclusions

After reviewing the existing methodologies to obtain numerical solutions for fluid dynamics problems, with their different degrees of accuracy and required computational efforts, this work will focus on wall-modeled LES and specifically, wall shear stress models. According to our opinion, WMLES has the most reasonable ratio between results accuracy and computational cost. It benefits from the accurate temporal and spatial resolution of an LES, in which valuable time-dependent data such as second-order statistics or vortex shedding frequencies is obtained, while the huge costs of resolving the boundary layer are fairly mitigated by the wall models. Additionally, unlike in the hybrid RANS/LES approach, the LES and the wall models are completely independent each other, offering broad flexibility when choosing the most suitable models for each domain and specific flow conditions.

To perform a successful LES computation, whether it is wall-resolved or wall-modeled, it is necessary to ensure some requirements such as the domain size sufficiency, the span of the computation in time, the required grid resolution in the wall and the far-field regions among others. In Chapter 2, a systematic methodology is presented in order the make an a priori and a posterior evaluation of the quality and reliability of a given LES simulation. Additionally, a new tool to assess the grid resolution suitability will be presented.

Afterward, in Chapter 3, a Two-Layer model in which the full incompressible Reynolds Average Navier-Stokes equations are used as the wall layer governing equations will be presented. As previously commented in the wall shear stress models

Table 1.2: Numerical strategies for Computational Fluid Dynamics classified according their degree of modeling (DOM) and computational cost (CC). Symbols legend: (Maximum) $\Uparrow \uparrow \nearrow \rightarrow \searrow \downarrow \Downarrow$ (Minimum). The table is split into methodologies suitable for free and wall-bounded flows.

Free flows

Methodology	**Applicabilty range**	**DOM**	**CC**
DNS	Any flow condition	$\Downarrow$	$\Uparrow$
LES	Any flow condition	$\downarrow$	$\uparrow$
RANS/URANS	Limited non-equilibrium flows	$\nearrow$	$\rightarrow$
Potential flow	inviscid and irrotational flows	$\Uparrow$	$\Downarrow$

Wall-bounded flows

Methodology	**Wall treatment**	**Wall model**	**Applicability range**	**DOM**	**CC**
DNS	N/A	N/A	Any flow condition	$\Downarrow$	$\Uparrow$
WRLES	N/A	N/A	Any flow condition	$\downarrow$	$\uparrow$
WMLES	Wall shear stress	URANS-based TLM	Non-equilibrium flows Detached boundary layer	$\searrow$	$\nearrow$
		TBLE-based TLM	Limited non-equilibrium flows Detached boundary layer	$\rightarrow$	$\rightarrow$
		integral BL	Limited non-equilibrium flows Detached boundary layer	$\rightarrow$	$\rightarrow$
		equilibrium TLM	Equilibrium flows Attached boundary layer	$\nearrow$	$\searrow$
		wall functions	Equilibrium flows Attached boundary layer	$\nearrow$	$\downarrow$
	Hybrid-RANS/LES	DES/DDES/iDDES	Non-equilibrium flows Detached boundary layer	$\rightarrow$	$\rightarrow$
RANS/URANS	N/A	N/A	Limited non-equilibrium flows Attached boundary layer	$\nearrow$	$\searrow$
	Wall shear stress	wall functions	Limited non-equilibrium flows Attached boundary layer	$\uparrow$	$\downarrow$
Potential flow	viscous boundary layer	N/A	Inviscid and irrotational Attached boundary layer	$\Uparrow$	$\Downarrow$

section, the TLM are affected by two recurrent problems, the "log-layer mismatch" (LLM) and the resolved Reynolds stresses (RRS) inflow. Until now, computationally expensive strategies have been proposed to deal with both issues separately[42, 44]. In the present formulation, a time-averaging filter (TAF) that tackles both problems at once with a single and low-computational-cost step is used for the first time in the TLM context, improving dramatically the wall model performance. The proposed wall model will be tested in equilibrium and non-equilibrium conditions with flows such as a Pipe flow case at $Re_\tau = 3000$, a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and full stall, and a fully 3D Ahmed car body at $Re = 7.56 \times 10^5$. Good numerical results are obtained in all operating conditions. The ability of the TAF in suppressing the LLM and blocking the RRS inflow simultaneously will be evaluated with the tests, showing that the new TLM methodology is a highly efficient technique compared to other existing implementations.

Then, in Chapter 4, the physical and mathematical formulation of the wall model presented in Chapter 3 is developed. The physical approach is based on the URANS equations whose temporal evolution is solved through an implicit scheme to avoid the CFL time step size restrictions. Additionally, the details of the numerical methodology used to obtain numerical solutions of the model are given.

In Chapter 5, the implementation details including the parallelization strategy will be explained. The implementation of a TLM model is of high complexity from a geometric and an algorithmic point of view. Since the purpose of a wall model is none other than reducing the computational cost of a given simulation, it makes no sense for the wall model to represent a significant part of the total computational efforts. Solving the URANS equations in the wall layer has a significant cost, and a good parallel strategy has to be implemented to make the most of the available computational resources. Given the significant complexity of the numerical structure required to solve the URANS equations, as well as the interaction process between the wall model and the LES domain, the implementation of a well-planned parallel strategy is far from straightforward. At the end of the chapter, the parallel efficiency is evaluated through a strong scalability test.

Finally, in Chapter 6, the conclusions obtained during the development of the present work are given. On the other hand, the future research is also outlined.

Bibliography

[1] R. Verstappen. When does eddy viscosity damp subfilter scales sufficiently? *J. Sci. Comput.*, 49:94–110, 2011.

[2] S. B. Pope. *Turbulent flows.* Cambridge Univesity Press, 2000.

[3] Y. Zhiyin. Large-eddy simultaion: Past, present and the future. *Chinese J. Aeronaut.*, 28:11–24, 2015.

[4] U. Piomelli. Wall-layer models for large-eddy simulations. *Prog. Aerosp. Sci.*, 44:437–446, 2008.

[5] J. Frölich. *Large Eddy Simulation turbulenter Strömungen.* Vieweg+Teubner Verlag, 2006.

[6] G. Alfonsi. Reynolds-averaged navier-stokes equations for turbulence modeling. *Appl. Mech. Rev.*, 62:040802, 2009.

[7] H. Schlichting and K. Gersten. *Boundary Layer Theory. 8th ed.* Springer-Verlag., 2004.

[8] L. Prandlt. Bericht über die Entstehung der Turbulenz. *Angew. Math. Mech.*, 5:136–139, 1925.

[9] T. Von Kármán. Mechanishe Ähnlichkeit und Turbulenz. *Nachrichten von der Gesellschaft der Wissenshaften zu Göttingen, Fachgruppe 1 (Mathematik).*, 5:58–76, 1930.

[10] L. Yarris. A flaw in the law. *Berkeley Lab:Highlights.*, pages 97–98.

[11] G.I. Barenblatt and V.M. Prostokishin. Scaling laws for fully developed turbulent shear flows. Part 2. Processing of experimental data. *J. Fluid Mech.*, 248:521–529, 1993.

[12] B. Cipra. A new theory of turbulence causes a stir among experts. *Science.*, 272:951, 1996.

[13] M.V. Zagarola, A.E. Perry, and A.J. Smits. Log laws or power laws: The scaling in the overlap region. *Phys. Fluids.*, 9:2094–2100, 1997.

[14] S. Hoyas and J. Jiménez. Scaling of the velocity fluctuations in turbulent channels up to $\mathrm{Re}_\tau = 2003$. *Phys. Fluids*, 18:011702, 2006.

[15] J.A. Sillero, J. Jiménez, and R.D. Moser. One-point statistics for turbulent wall-bounded flows at Reynolds numbers up to $\delta^+ \approx 2000$. *Phys. Fluids*, 25:105102, 2013.

[16] J. Ahn, J. H. Lee, J. Lee, J. H. Kang, and H. J. Sung. Direct numerical simulation of a 30R long turbulent pipe flow at $\mathrm{Re}_\tau = 3008$. *Phys. Fluids*, 27:065110, 2015.

[17] D. R. Chapman. Computational aerodynamics, development and outlook. *AIAA J.*, 17:1293–1313, 1979.

[18] H. Choi and P. Moin. Grid-point requirements for large eddy simulation: Chapman's estimates revisited. *Phys. Fluids*, 24:011702, 2012.

[19] Q. Wang, S.A. Gomez, P.J. Bloniganc, A.L. Gregoryd, and E.Y. Qiane. Towards scalable parallel-in-time turbulent flow simulations. *Phys. Fluids*, 25:110818, 2013.

[20] R.W.C.P. Verstappen and A.E.P. Veldman. Direct numerical simulation of turbulence at lower costs. *J. Eng. Math.*, 37:143–159, 1997.

[21] J. W. Deardorff. A numerical study of three-dimensional turbulent channel flow at large Reynolds numbers. *J. Fluid Mech.*, 41:453–480, 1970.

[22] F. X. Trias and O. Lehmkuhl. A self-adaptive strategy for the time-integration of Navier-Stokes equations. *Numer. Heat Tr. B-Fund.*, 60-2:116–134, 2011.

[23] P. R. Spalart. Detached-eddy simulation. *Annu. Rev. Fluid Mech.*, 41:181–202, 2009.

[24] U. Schumann. Subgrid-scale model for finite difference simulation of turbulent flows in plane channels and annuli. *J. Comput. Phys.*, 18:376–404, 1975.

[25] G. Groetzbach. Direct numerical and large eddy simulation of turbulent channel flows. *Encyclopaedia of Fluid Mechanics, Gulf Publishing, Houston, Texas.*, 6:1337–1391, 1987.

[26] U. Piomelli, P. Moin, J. Ferziger, and J. Kim. New approximate boundary conditions for large eddy simulations of wall-bounded flows. *Phys. Fluids*, 101:61–68, 1989.

[27] G. Hoffman and C. Benocci. Approximate wall boundary conditions for large eddy simulations. *Advances in Turbulence V, Fluid Mechanics and its Applications*, 24:222–228, 1995.

[28] L. Temmerman, M. A. Leschziner, and K. Hanjalic. A-priori studies of a near-wall rans model within a hybrid les/rans scheme. In W Rodi and N Fueyo, editors, *Engineering Turbulence Modelling and Experiments 5*, pages 317–326, 2002.

[29] H. Werner and H. Wengle. Large-eddy simulation of turbulent flow over and around a cube in a plate channel. In *Turbulent Shear Flows*, volume 8. Springer-Verlag, 1993.

[30] E. Balaras and C. Benocci. Subgrid-scale models in finite-difference simulations of complex wall bounded flows. In *AGARD CP 551*, pages 2.1–2.5, 1994.

[31] E. Balaras, C. Benocci, and U. Piomelli. Two-layer approximate boundary conditions for large-eddy simulations. *AIAA J.*, 34 (6):1111–1119, 1996.

[32] W. Cabot. Near-wall models in large-eddy simulations of flow behind a backward-facing step. *Annual Research Brief. Center for Turbulence Research, Stanford, CA*, pages 199–210, 1996.

[33] W. Cabot and P. Moin. Approximate wall boundary conditions in the large-eddy simulation of high Reynolds number flow. *Flow Turbul. Combust.*, 63:269–291, 1999.

[34] M. Wang and P. Moin. Dynamic wall modelling for large-eddy simulation of complex turbulent flows. *Phys. Fluids*, 14(7):2043–2051, 2002.

[35] F. Tessicini, L. Temmerman, and M. A. Leschziner. Approximate near-wall treatments based on zonal and hybrid RANS-LES methods for LES at high Reynolds numbers. *Int. J. Heat Fluid Fl.*, 27(5):789–799, 2006.

[36] F. Tessicini, N. Li, and M. A. Leschziner. Large-eddy simulation of three-dimensional flow around a hill-shaped obstruction with a zonal near-wall approximation. *Int. J. Heat Fluid Fl.*, 28:894–908, 2007.

[37] S. Kawai and K. Asada. Wall-modeled large-eddy simulation of high Reynolds number flow around an airfoil near stall condition. *Comput. Fluids*, 85:105–113, 2013.

[38] V. G. Diurno, E. Balaras, and U. Piomelli. Wall-layer models for LES of separated flows. *Modern Simulation Strategies for Turbulent Flows, ed. B. Geurts. Philadelphia, PA: RT Edwards*, pages 207–222, 2001.

[39] P. R. Spalart and S. Allmaras. A one-equation turbulence model for aerodynamic flows. *La Recherche Aerospatiale*, 1:5–21, 1994.

[40] U. Piomelli and E. Balaras. Wall-layer models for large-eddy simulations. *Annu. Rev. Fluid Mech.*, 34:349–374, 2002.

[41] S. Kawai and J. Larsson. A dynamic wall model for large-eddy simulations of high Reynolds number compressible flows. *Annual Research Brief. Center for Turbulence Research, Stanford, CA*, pages 25–37, 2010.

[42] S. Kawai and J. Larsson. Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy. *Phys. Fluids*, 24:015105, 2012.

[43] S. Kawai and J. Larsson. Dynamic non-equilibrium wall-modeling for large eddy simulation at high Reynolds number. *Phys. Fluids*, 25:015105, 2013.

[44] G. I. Park and P. Moin. An improved dynamic non-equilibrium wall-model for large eddy simulation. *Phys. Fluids*, 26:015108, 2014.

[45] S. Patil and D. Tafti. Wall modeled large eddy simulations of complex high Reynolds number flows with synthetic inlet turbulence. *Int. J. Heat Fluid Fl.*, 33:9–21, 2012.

[46] J. Bodart and J. Larsson. Wall-modeled large-eddy simulation in complex geometries with application to high-lift devices. *Annual Research Brief. Center for Turbulence Research, Stanford, CA*, pages 37–48, 2011.

[47] J. Bodart and J. Larsson. Sensor-based computation of transitional flows using wall-modeled large-eddy simulation. *Annual Research Brief. Center for Turbulence Research, Stanford, CA*, pages 229–240, 2012.

[48] J. Larsson, S. Kawai, J. Bodart, and I. Bermejo-Moreno. Large-eddy simulation with modeled wall-stress: recent progress and future directions. *Mech. Engng. Rev.*, 3(1):1–23, 2016.

[49] P. R. Spalart, W. H. Jou, M. Strelets, and S. R. Allmaras. Comments on the feasibility of LES for wings and on a hybrid RANS/LES approach. advances in DNS/LES. In *1st AFOSR Interational Conference on DNS/LES*, number 137–148. Greyden Press, 1997.

[50] F. R. G. Menter and M. Kuntz. Adaptation of eddy-viscosity turbulence models to unsteady separated flow behind vehicles. *The Aerodynamics of Heavy Vehicles: Trucks, Buses, and Trains*, 19:339–352, 2004.

[51] M. Strelets. Detached eddy simulation of massively separated flows. In *39th Aerospace Sciences Meeting and Exhibit.*, number 1–18. American Institute of Aeronautics and Astronautics., 2001.

[52] N. V. Nikitin, F. Nicoud, B. Wasistho, K. D. Squires, and P. R. Spalart. An approach to wall modeling in large-eddy simulations. *Phys. Fluids*, 12:1629, 2000.

[53] U. Piomelli, E. Balaras, H. Pasinato, K. D. Squires, and P. R. Spalart. The inner-outer layer interface in large-eddy simulations with wall-layer models. *Int. J. Heat Fluid Flow*, 24:538–550, 2003.

[54] J. Larsson, F. S. Lien, and E. Yee. Feedback-controlled forcing in hybrid les/rans. *Int. J. Comput. Fluid Dyn.*, 20:687–699, 2006.

[55] P. R. Spalart, S. Deck, M. L. Shur, K. D. Squires, M. Kh. Strelets, and A. Travin. A new version of detached-eddy simulation, resistant to ambigous grid densities. *Theor. Comput. Fluid Dyn.*, 20:181–195, 2006.

[56] M. L. Shur, P. R. Spalart, M. Kh. Strelets, and A. K. Travin. A hybrid rans-les approach with delayed-des and wall-modelled les capabilities. *Int. J. Heat Fluid Flow*, 29:1638–1649, 2008.

[57] G. Kalitzin, G. Medic, G. Iaccarino, and P. Durbin. Near-wall behavior of RANS turbulence models and implications for wall functions. *J. Comput. Phys.*, 204:265–291, 2005.

[58] W. Vieser and F.R. Esch, T. Menter. Heat transfer predictions using advanced two-equation turbulence models. *CFX Technical Memoranum CFX-VAL10/0602*, 2002.

[59] M Drela and M.B. Giles. Viscous-inviscid analysis of transonic and low reynolds number airfoils. *AIAA J.*, 25(10), 1987.

[60] M. Drela. An integral boundary layer formulation for blunt trailing edges. *AIAA J.*, 25(10), 1989.

2

Quality assessment of high fidelity CFD simulations

2.1 Introduction.

The movement of a fluid is a very complex and varied phenomenon, being extremely difficult to define a general methodology to perform a successful numerical simulation of it. There is a vast range of different physical phenomena that can be triggered within a given fluid domain depending on the boundary geometries and characteristics, the flow velocity and the physical properties of the fluid itself. To ensure the quality of a high fidelity CFD simulation, several aspects have to be taken into account. A key issue for any accurate flow simulation is the sufficiency of the computational domain size, which has to be able to accommodate the biggest flow structures within its boundaries. If this requirement is not met, the largest energy containing flow structures will be missed from the simulation, and thus, their effects on the flow too. Analogously, the simulated temporal period has to be large enough to represent the largest flow structures. The simulation span should be at least of the same order of magnitude as the largest flow motion characteristic timescale. On the other hand, other numerical aspects have to be also taken into account, such as the grid resolution both, in the wall area and the far field. In this Chapter, a new systematic methodology to assess the computational grid resolution sufficiency is presented. Regarding the temporal resolution, if the CFL condition is followed, a sufficient temporal accuracy is ensured [1, 2].

Unfortunately, all the simulation parameters detailed above are characteristic of each particular flow, and there is no possibility to establish a general methodology to set them up without an a priori knowledge of the flow behavior. Nonetheless, this previous information is not always available, and an initial simulation has to

performed to estimate the computational parameters. Afterward, the simulation variables have to be readjusted according to the information obtained through the methodologies detailed below.

A summary of key aspects and parameters that have to be appropriately set up in a temporal and spatial accurate simulation is presented in table 2.1. The methodologies or physical magnitudes used to evaluate their correct configuration are also detailed.

Table 2.1: Computational parameters to be considered for a high fidelity CFD simulation set up. On the right, flow magnitudes and statistical tools used for the parameters configuration assessment.

Simulation parameter	Control method or magnitude
Computational domain size	Two-point correlation $\Re_{\phi\phi}$
Initial transient period	Temporal evolution of a given variable
Temporal simulation span	Frequency spectrum of a temporal signal of a given variable
Grid resolution in the far field	Ratio between the grid size and the Kolmogorov scales (η)
Grid resolution in the near-wall region	First off-wall node distance and grid spacings in wall-parallel directions

Before detailing the different methodologies intended to assess the reliability of a given time and space-resolved CFD simulation, several physical quantities and statistical methods useful for the quality assessment process will be introduced. These magnitudes and methodologies are the two-point correlation function, the flow energy spectrum and the second invariant of the velocity gradient tensor.

2.2 Statistical methods and flow quantities for quality assessment.

2.2.1 The two-point correlation and the energy spectrum

The two-point, one-time autocovariance tensor $\Re_{ij}(\boldsymbol{r}, \boldsymbol{x}, t)$ is defined as [3]:

$$\Re_{ij}(\boldsymbol{r}, \boldsymbol{x}, t) = \langle u_i(\boldsymbol{x}, t) u_j(\boldsymbol{x} + \boldsymbol{r}, t) \rangle, \tag{2.1}$$

where u_i is the i_{th} velocity component, $\boldsymbol{x} = (x, y, z)$ is a spatial position within the flow domain, and $\boldsymbol{r}$, is the position vector between $\boldsymbol{x}$ and a second domain point. This expression is more commonly known as two-point correlation tensor, and it is the simplest statistical magnitude containing information about the spatial structure of the flow. For homogeneous isotropic turbulence, the two-point correlation is independent of $\boldsymbol{x}$, and since the mean velocity is zero, the instantaneous velocity field is equal to its fluctuating component ($u_i = u_i'$). Taking the Fourier transform of $\Re_{ij}(\boldsymbol{r}, t)$, we obtain the energy spectrum tensor $\Phi_{ij}(\boldsymbol{\kappa}, t)$:

$$\Phi_{ij}(\boldsymbol{\kappa}, t) = \frac{1}{(2\pi)^3} \iiint_{-\infty}^{\infty} e^{-ik\boldsymbol{r}} \Re_{ij}(\boldsymbol{r}, t) d\boldsymbol{r}, \tag{2.2}$$

The $\Re_{ij}(\boldsymbol{r}, t)$ and $\Phi_{ij}(\boldsymbol{\kappa}, t)$ tensors contain the same information about the flow field although expressed in two different spaces, the spatial and the frequency space, respectively.

The energy spectrum tensor is a relevant magnitude since it measures the kinetic energy content of eddies with wavenumber (a measure of the characteristic time-scale) κ. Derived from the $\Phi_{ij}(\boldsymbol{\kappa}, t)$ tensor, for homogeneous isotropic turbulence, the energy spectrum function $E(\kappa, t)$ can be defined as:

$$E(\kappa, t) = \iiint_{-\infty}^{\infty} \frac{1}{2} \Phi_{ii}(\boldsymbol{\kappa}, t) \delta(|\boldsymbol{\kappa}| - k) d\boldsymbol{\kappa}, \tag{2.3}$$

This function may be useful to understand better the physical data contained in $\Phi_{ij}(\boldsymbol{\kappa}, t)$. Actually, the $E(\kappa, t)$ function contains the same data than the energy spectrum tensor but devoid of all directional information. The main property of the energy spectrum function is that its integral over the whole range of wavenumbers (fluid motion scales) accounts for the total amount of kinetic energy (for homogeneous isotropic turbulence is equal to the turbulent component) contained in the flow per unit mass. For a specific instant of time, it is expressed as:

$$\int_0^{\infty} E(\kappa) dk = \frac{1}{2} \langle u_i u_i \rangle, \tag{2.4}$$

In figure 2.1, a schematic plot of the energy spectrum function is displayed. The spectrum is split into three main regions: The energy-containing range which contains the low wavenumber (largest) structures which receive the external kinetic energy inflow, the inertial subrange in which the flow structure break-up process occurs transferring kinetic energy from the large to the small structures (from lower to higher wavenumbers), and finally, the dissipation range in which the smaller structures (highest wavenumbers) vanish due to viscous effects, converting the turbulent kinetic energy into thermal energy. Taking into account the Kolmogorov similarity hypothesis [3], the behavior of the inertial range is considered universal (i.e., the energy cascade process from the largest to the smallest scales) and can be described as:

$$E(\kappa) = C_k \varepsilon^{2/3} \kappa^{-5/3} \tag{2.5}$$

where C_k is the universal Kolmogorov constant, and ε is the dissipation rate. The rate at which the turbulent kinetic energy is dissipated by viscous effects can be also obtained for $E(\kappa)$:

$$\varepsilon = \int_0^\infty 2\nu\kappa^2 E(\kappa) dk, \tag{2.6}$$

coming most of the dissipative contribution from the highest wavenumbers as the integral κ^2 dependency shows.

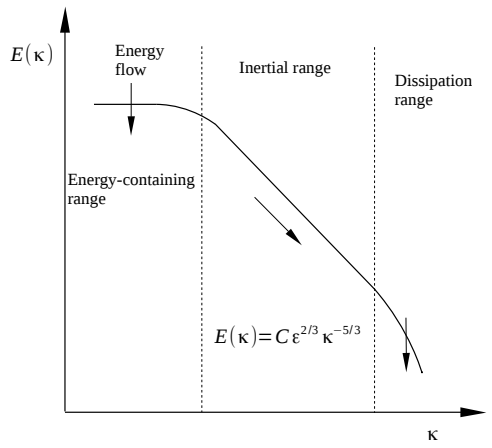

Figure 2.1: Turbulent kinetic energy spectrum for homogeneous isotropic turbulence. The main three ranges, energy containing, inertial and dissipative subrange are shown.

2.2.2 Methods for fluid structure identification

A fluid domain is a continuum in which the events occurring within it are not straightforwardly revealed. In order to gain insight into the real flow behavior, different analysis techniques of physical properties or flow magnitudes have to be used. The energy power spectrum is one of the existing approaches but the information it provides, although being extremely valuable, it is somewhat abstract and not very intuitive. One of the difficulties in fluid dynamics is the identification and visualization of the different structures contained in the flow domain. By directly visualizing the flow motions, the laminar and turbulent regions could be easily located and the relevant flow structures identified. In order to recognize a fluid structure, a distinctive characteristic feature of it must be identified in order to isolate a given motion from the rest of the flow field. Nonetheless, this is not straightforward at all since even the definition of vortex remains an open question [4]. For instance, Lugt [5] defines a vortex as the rotating motion of a multitude of material particles around a common center. This definition describes a vortex as an isovorticity region in which the streamlines are closed and features a pressure minimum in its rotation center. Nevertheless, spiraling streamlines or pathlines can be only observed if an observer moves along with the vortex itself, and the existence of a local pressure minimum does not guarantee the existence of a vortex.

Many different vortex identification methods are using different flow quantities, each one with its strengths and weaknesses. However, the structure shape and behavior obtained with a given method will have to be interpreted according to the physical meaning of the property or magnitude used in the analysis. A complete discussion on visualization methodologies can be found in the review published by Kolár [4] in 2007. In the present work, the focus will be placed on the methods based on the analysis of the velocity gradient tensor $\nabla \boldsymbol{u}$, since they provide a rational basis for vortex identification and the general classification of 3D flows [6].

Firstly, a short introduction to tensor invariants will be given to better understand the physical interpretation of the $\nabla \boldsymbol{u}$ analysis. Afterward, the methodology based on the second invariant of $\nabla \boldsymbol{u}$ commonly known as Q-criterion will be discussed.

Tensor invariants

In linear algebra, the invariants of a tensor are the coefficients of the characteristic polynomial of a given $n \times n$ tensor $\boldsymbol{A}$:

$$p(\lambda) = det(\boldsymbol{A} - \lambda \boldsymbol{I}) \longrightarrow \lambda^n + I_A \lambda^{n-1} + II_A \lambda^{n-2} + ... + (N-1)_A \lambda + N_A = 0, \quad (2.7)$$

where $\lambda \in \mathbb{C}$, $\boldsymbol{I}$ is the identity tensor, and $I_A, II_A, ..., N-1_A, N_A$ are the N invariants of $\boldsymbol{A}$. As the name implies, their value does not depend on the $\boldsymbol{A}$ reference frame.

In engineering, most used tensors are of 3×3 size. For this particular case, the characteristic polynomial can be written as:

$$p(\lambda) = \lambda^3 - P\lambda^2 + Q\lambda - R = 0, \tag{2.8}$$

where P, Q, and R, are the first, second, and third invariants of tensor $\boldsymbol{A}$ which can be expressed as:

$$P = tr(\boldsymbol{A}), \tag{2.9}$$

$$Q = \frac{1}{2}(tr(\boldsymbol{A})^2 - tr(\boldsymbol{A}^2)), \tag{2.10}$$

$$R = det(\boldsymbol{A}), \tag{2.11}$$

Tensor invariants are widely used in engineering since they contain relevant physical information while keeping its value independent from the coordinate system position. Therefore, any function expressed in terms of tensor invariants will also be independent of coordinate axis rotations, allowing the construction of general expressions for the related physical phenomenon.

In fluid dynamics, the velocity gradient tensor is a relevant tensorial variable, containing valuable information about the flow behavior and structure. Therefore, its invariants will contain a part of this data.

The $\nabla \boldsymbol{u}$, as a second-order tensor can be decomposed into its symmetric and its antisymmetric part. In Einstein notation, the decomposition can be written as:

$$G_{ij} = \frac{\partial u_i}{\partial u_j} = S_{ij} + \Omega_{ij}, \tag{2.12}$$

where G_{ij} is the velocity gradient tensor, S_{ij} is the rate-of-strain tensor, and Ω_{ij} is the rate-of-rotation or vorticity tensor. Both, S_{ij} and Ω_{ij} are key magnitudes containing all the scalar and directional information about flow deformation and rotation, and can be expressed as:

$$S_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial u_j} + \frac{\partial u_j}{\partial u_i}\right) \quad ; \quad \Omega_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial u_j} - \frac{\partial u_j}{\partial u_i}\right), \tag{2.13}$$

Therefore, the G_{ij} invariants will also contain valuable information about the flow deformation and rotation, but stripped from any directional data.

The first invariant of G_{ij}, is the trace of the tensor:

$$P = tr(G_{ij}) = \frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} + \frac{\partial w}{\partial z}, \tag{2.14}$$

where u, v and w are the three velocity components. Therefore, the first invariant of G_{ij}, is the divergence of the velocity field that for incompressible flows is zero.

Regarding the second invariant, it can be expressed as:

$$Q = \frac{1}{2}\left(tr(G_{ij})^2 - tr(Gij^2)\right) = \frac{1}{4}||\omega_{ij}||^2 - \frac{1}{2}||S_{ij}||^2, \tag{2.15}$$

where ω_{ij} is the vorticity vector defined as $\omega_{ij} = \epsilon_{ijk}\partial_j u_k = \nabla \times \boldsymbol{u}$, and $||\cdot||$ is the norm operator for either a vector or tensor. Therefore, according this expression, the Q invariant represents the local balance between shear strain rate and vorticity.

The Q-criterion

Since the vorticity tensor represents the angular velocity of fluid elements and has invariant properties, it may seem an appropriate magnitude for a vortex identification. Nonetheless, many authors have found that vorticity is not a suitable magnitude for this purpose since it cannot distinguish between pure shearing motions and the swirling behavior of a vortex [7, 8, 9]

By contrast, the Q-criterion defines a vortex as a connected fluid region with a positive second invariant Q [10]. Considering the definition of the second invariant Q in equation 2.15, we could say that Q represents the local balance between the shear strain rate and the vorticity magnitude. Therefore, according to this physical interpretation, vortices are defined as areas where the vorticity magnitude is greater than the magnitude of rate-of-strain [4, 10]. As a consequence, the deficiencies presented by the vorticity as a vortex identification magnitude are corrected since the shear-dominant effects are also considered. Additionally, the Q-criterion requires the pressure in the vortex region to be lower than the pressure in its surrounding area [4].

In figure 2.2, an example of flow structure visualization through isosurfaces of Q-criterion is shown. In this example, the flow past a FX77-W-500 airfoil at $Re = 2.75 \times 10^6$ at $AoA = 0^\circ$ is depicted. In the plot, quasi-laminar (shear-dominant) regions on the upper surface of the airfoil are revealed, while in the wake region, large Von-Kármán vortex together with smaller turbulent (vorticity-dominant) flow structures are displayed.

2.3 Computational domain size

The size of the computational domain is a critical aspect that has to be correctly addressed. Any flow structure larger than the selected domain size will not be represented in the flow simulation, and therefore, its effects will not be taken into account. This may lead to substantial errors in the numerical predictions and must be minimized. On the other hand, in many cases, the enlargement of the computational

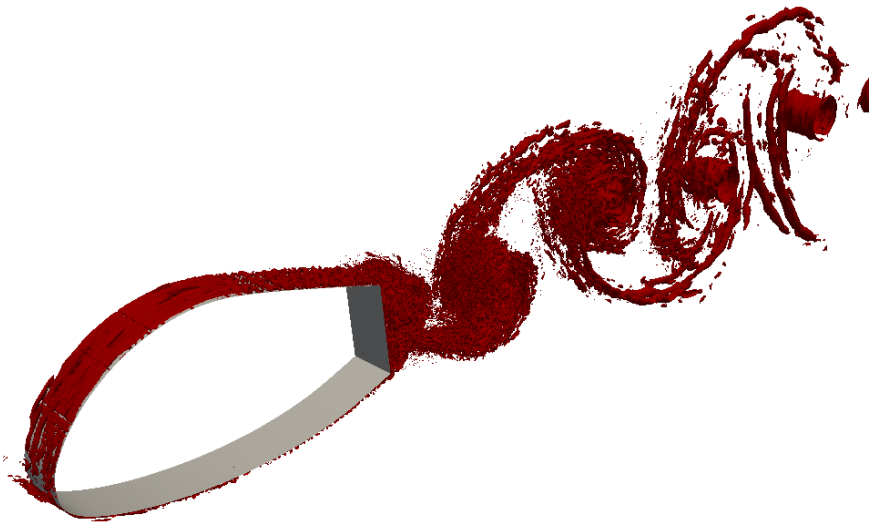

Figure 2.2: Isosurfaces of the second invariant of the velocity gradient tensor for a FX77-W-500 airfoil at $Re = 2.7 \times 10^6$ and $AoA = 0°$.

domain may entail an unaffordable increase of the computational cost, especially if solid boundaries are extended. Therefore a compromise between the numerical results accuracy and the required computational resources must be found.

To evaluate whether a given domain size is sufficient to accommodate all the characteristic flow structures, the normalized two-point correlation function, which is conceptually the same of the two-point correlation tensor explained in section 2.2.1 but for a fluctuating velocity component in a specific direction, will be used. This methodology was initially proposed for this purpose by Comte-Bellot in 1963[11]. The two-point correlation function gives the statistical correlation between two random variables representing the same quantity measured at two different points separated by a given distance in space or time. This magnitude is a useful indicator of dependency of the values of a given variable measured in two points separated in space or time, and it can be used to assess the distance required between two sample points for the function values to be effectively uncorrelated.

In high fidelity computational fluid dynamics, this statistical tool is especially useful given the nature of the flow structure commented in Section 2.2.1. If we consider a given spatial position placed in a fully turbulent flow region, the instantaneous velocity at this point will be an overlap of velocities derived from the different structures containing this point. In Figure 2.3, this effect is depicted. The velocity $\mathbf{u}_1$ is a composition of velocities derived from the belonging of p_1 to the structures s_1, s_2 and s_3. On the other hand, the velocity $\mathbf{u}_2$ is only affected by the structure $s1$. Although the temporal signals of $||\mathbf{u}_1||$ and $||\mathbf{u}_2||$ will apparently be significantly different, their fluctuations will be statistically correlated through the contribution of the s_1 motion which contains both points at the same time.

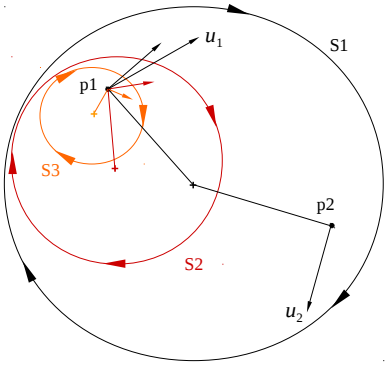

Figure 2.3: Sketch of the velocity components at two different spatial positions (p_1 and p_2). The total velocity at a given spatial point, is the summation of velocity contributions from each flow structure containing the point. s_1, s_2 and s_3 are three different flow structures while $\mathbf{u_1}$ and $\mathbf{u_2}$ are total velocity vectors at p_1 and p_2, respectively.

Therefore, if the velocity signals of two spatially separated points are statistically uncorrelated, it means that there is no flow structure including them. As a consequence, no flow structure in the region is sufficiently large to contain both points at once, and therefore, it will be smaller than the distance between the two spatial positions.

The normalized two-point correlation of the ϕ velocity component fluctuations ($\Re_{\phi\phi}$) in a given spatial direction $\delta\boldsymbol{r}$ is evaluated according the following expression [3]:

$$\Re_{\phi\phi}(\boldsymbol{x}, \delta\boldsymbol{r}) = \frac{\langle \phi'(\boldsymbol{x}, t)\phi'(\boldsymbol{x} + \delta\boldsymbol{r}, t)\rangle}{\langle \phi'^2\rangle}, \tag{2.16}$$

where ϕ' is the fluctuation of any of the three velocity components, $\boldsymbol{x} = (x, y, z)$, and $\langle\cdot\rangle$ denotes averaging over time.

The primary purpose of this method is to determine the maximum size of any flow structure existing within the computational domain. Hence, for this method to work, the probes to obtain the signal for computing the statistical correlation, must be placed in the areas where the largest flow structures are located. If no previous information on the flow is available, Q isosurface plots must be used to identify those areas, ensuring that the probes have been appropriately positioned [10, 12].

A good example to illustrate the use of this methodology is the simulation of flows with homogeneous periodic directions. These flows are characterized by having at least one direction in which the orthogonal cross-section geometry remains constant, for instance, the extrusion of a two-dimensional (2D) shape such as a cylinder or a 2D airfoil. Other examples could be internal flows such as a pipe or channel flows. The periodic condition is used to emulate an infinite domain length in the homogeneous direction by connecting the boundary-contiguous nodes of one side of the other periodic direction with the nodes of the other side. In the case of Pipe flow, this strategy would emulate the simulation of a toroidal domain with a perimeter length equal to the original pipe length as sketched in figure 2.4 but without considering centrifugal effects.

In those cases, any flow structure larger than the periodic distance will not be represented in the simulation since it would be overlapped with itself through the periodic boundaries. In order to avoid any unphysical misrepresentation, the two-point correlation function between the two periodic planes must be evaluated to confirm that it does not exist any flow structure larger than the periodic extrusion length. An example of an application of this technique can be found in the work of Pont-Vílchez et al. [13]. A DNS of a Backward-facing step at Reynolds $Re_\tau = 395$ was computed in order to make an extensive analysis of the characteristics of this canonical flow. In this case, the cross-section geometry remains constant in the spanwise direction, and a periodic condition can be used to emulate an infinite domain in the cross-flow direction. On the other hand, in the streamwise direction, the case geometry features a sudden expansion which causes a boundary layer detachment and a large flow recirculation right after the step (see Figure 2.5).

In order to ensure the reliability of the DNS simulation, it must be confirmed that the domain spanwise length is sufficient to contain the largest flow structures in that direction. A set of probes were placed throughout the computational domain, putting special emphasis in the recirculation region where the largest coherent structures are expected to be found. In figure 2.5, the different probe positions, from $P1$ to

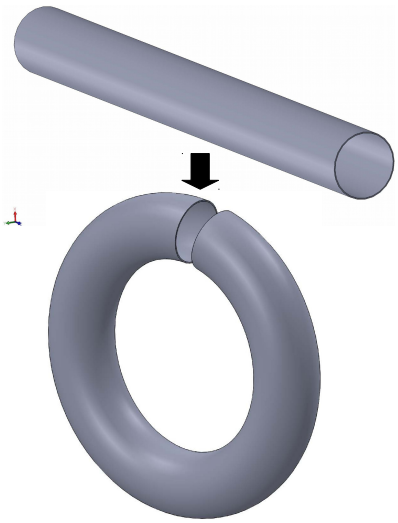

Figure 2.4: Pipe flow geometry. The periodic condition used in the streamwise direction emulates the flow within a toroidal ring without inertial effects.

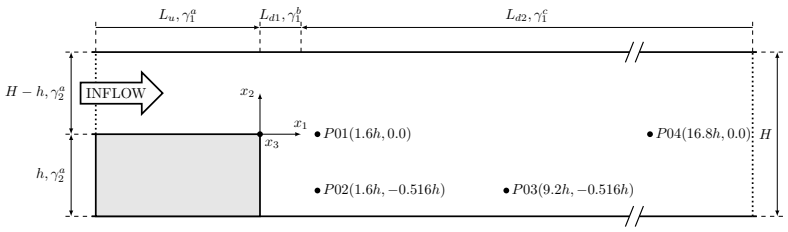

Figure 2.5: Span-normal cross-section of the Backward-facing step case. *P*0*i* are the positions of the numerical probes used to obtain the temporal signal of the flow variables [13].

*P*4, are displayed. It must be noticed that computing a two-point correlation in a laminar region without turbulent activity does not make sense since, by definition, no turbulent flow structures nor velocity fluctuations do exist. In Figure 2.6, the normalized two-point correlation of the streamwise velocity component, which is the most energetic and relevant direction, is displayed along the periodic direction for the four probe positions.

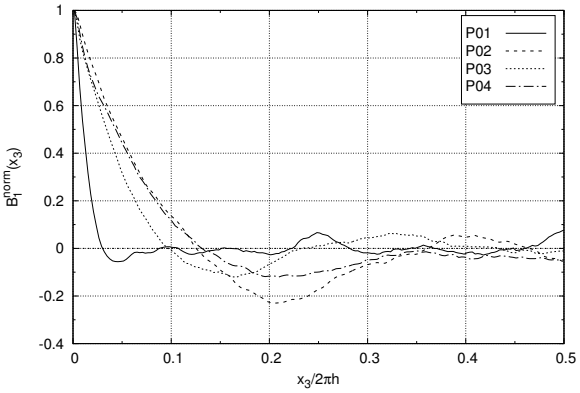

Figure 2.6: Two-point correlation of the streamwise velocity component fluctuations along the spanwise direction. The signals are obtained in the four different probes displayed in figure 2.5 [13].

According to the two-point correlation results in Figure 2.6, the spanwise distance of the simulation is sufficient to reproduce the whole range of flow structures since the velocities along the periodic direction, become uncorrelated before reaching half of the domain spanwise length. That indicates that in the periodic direction, all the flow structures are smaller than the extrusion distance and thus, they are well represented. In the $P1$ case, the correlation tends to zero significantly faster than in the other probes. At this position, the boundary layer is suddenly detached from the step's upper surface due to the domain expansion. The breakup process that suffers the boundary layer in this region probably leads to small structures that might be recomposed in larger eddies downstream.

It is worth to comment that neglecting the largest flow structures are not always affecting all the flow magnitudes equally. Lozano-Durán and Jiménez [14] performed computations of Channel flow cases with different domain sizes and different Reynolds numbers. They concluded that some magnitudes were affected by the use of excessively small computational domains while other variables remained unaffected. While the first-order statistics such as the mean velocity profiles remained unchanged when reducing the computational box size, the second-order statistics (i.e., the rms velocity fluctuations), were significantly underpredicted for the smaller domain. Enlarging the computational box for a Channel flow can make the numerical simulation unfeasible since the wall area is significantly increased. Therefore, when choosing the domain dimensions, it must be considered what kind of flow data are

we interested in obtaining and the desired degree of accuracy. Nonetheless, in most flows, the consequences of neglecting the largest structures are unknown.

Finally, it is necessary to remark that the two-point correlation function requires a significant amount of temporal data to be computed. If a priori information of the flow is not available, this forces to perform an initial computation, setting up the simulation parameters with roughly estimated values. Once a first numerical data is obtained, the parameters can be fine-tuned for a second simulation.

2.4 Initial transient period

Four main aspects condition a given flow configuration. The computational domain size and geometry, the boundary conditions, the flow motion source which can be imposed through the boundary condition or a momentum source term, and finally, the fluid physical properties. As previously commented, a vast range of different flow conditions and features may be engendered within the fluid domain depending on the conditioning factors listed above. Frequently, the flow may have apparent chaotic behavior. However, if the commented parameters remain constant, a temporally periodic pattern can be found in the flow motion. This flow condition is called the statistically stationary regime in which even the largest flow structures are cyclically repeated along time.

As its name suggests, the statistically stationary regime entails that a constant mean flow pattern can be obtained by averaging the instantaneous flow variables in time. The mean flow values are the ones used to compare two different flows, or comparing the results with a reference for validation, since comparing the instantaneous flow field has little sense. It is highly unlikely to find significant similitudes when comparing two random instantaneous turbulent flow fields, even from the same flow conditions. Nonetheless, if the flow field is averaged in time for a sufficiently long period, the mean flow field should be identical.

Even so, the statistically stationary regime is not instantaneously reached. Depending on the initial conditions of the simulation, a transient period must be simulated before reaching the periodic state. The farther the initial conditions from the real flow conditions, the longer the necessary time lapse. In many cases, the computational cost of the initial transient period can be very expensive, and when possible, it is highly recommended to use techniques to shorten it. For instance, if a laminar flow field is used to initialize a turbulent flow, random numerical perturbations can be initially introduced in order to trigger the turbulent behavior quicker.

To obtain well-converged mean flow statistics, the averaging process must be started when the statistically stationary regime is reached, and thus, the initial transient period must be skipped. Hence, it is necessary to identify the onset of the periodic stage. To do so, a monitoring probe must be placed in the domain region

where the most relevant flow structures are located, by either, checking previously known flow data or Q-isosurface plots. By analyzing the temporal signal of the different problem variables, it can be observed whether their signal oscillates about a mean value. Other magnitudes can be monitored provided that they are representative of the most relevant flow structures. For instance, the instantaneous lift coefficient signal in a flow past an airfoil at full stall can be used to determine the inception of the cyclical period. In this kind of flows, the boundary layer detachment/reattachment process is the most relevant event, and it strongly influences the value of the lift coefficient. Therefore, its temporal signal will provide information about the evolution of the global behavior.

In Figure 2.7, the temporal signal of the lift coefficient (C_l) of a DU 91-W2-250 airfoil in full stall condition is shown. In this specific case, the simulation was initialized with a uniform velocity field throughout the computational domain which is non-physical, and thus, substantially different from the statistically stationary regime in the airfoil vicinity and wake. It can be easily identified the transient period from $t = 0$ up to $t \approx 40$ and from then, how the lift signal starts to fluctuate about an average value with quasi-cyclical periods.

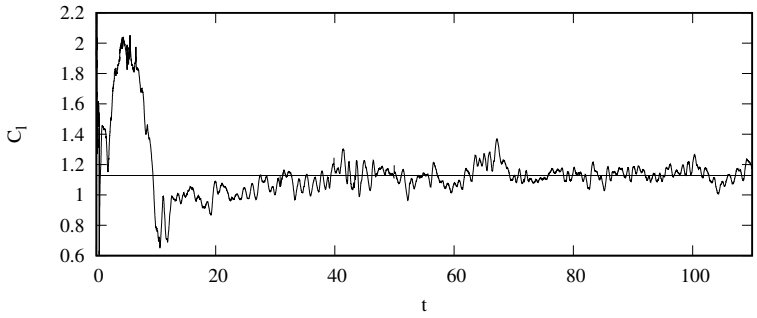

Figure 2.7: Instantaneous lift coefficient signal of a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and stall condition. [15]

2.5 Time average period

Once the initial transient period is identified, it is necessary to determine the necessary averaging period length to obtain well-converged mean flow statistics. Hence, the characteristic timescale of the largest flow structure must be determined since it will indicate the minimum value of the averaging period span. To do so, the frequency

spectrum of the temporal signals obtained in the previous step will be used in order to reveal the frequency of the different flow structures passing through the regions where the different probes are located.

Since the temporal signal of a velocity component contains information about the spatial structures present in probe location (see Section 2.3), when performing its Fourier transform, the frequency spectrum obtained is similar to the power spectra detailed in Section 2.2.1, provided that the probe is located in a developed turbulent region. In Figure 2.8, the power spectra of a streamwise velocity signal obtained in the center of a fully developed turbulent Channel flow is depicted. In the image, the characteristics of an energy power spectrum can be easily identified, such as the $\kappa^{-5/3}$ slope of the inertial range and the dissipation region at the highest frequencies.

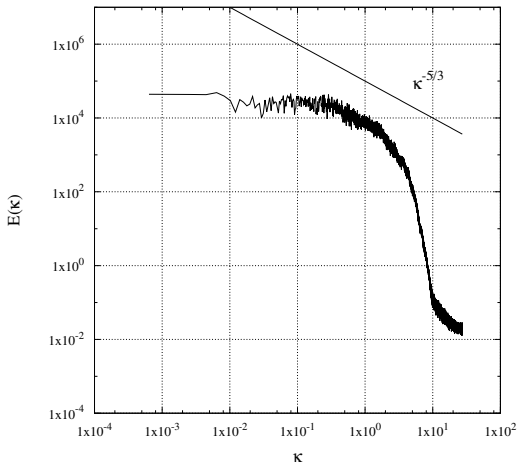

Figure 2.8: Power spectrum function of a streamwise velocity component signal obtained at the center of a fully developed turbulent Channel flow at $Re_\tau = 2000$.

However, the power spectrum characteristic shape depicted in Figure 2.1 is valid for homogeneous isotropic turbulence in which the energy cascade is the only relevant process occurring within the flow. In a more complex flow, the turbulent activity itself can be overlapped with other flow features such large vortical structures or boundary layer detachments among others. All these phenomena will contribute to the temporal signals of almost any quantity related to the flow, especially to the velocity components.

When computing the power spectra of the u_i temporal signal, the characteristic frequencies of all the flow structures contributing to the temporal wave will be

revealed in the form of a peak in the spectrum curve. The height of the peak will be a measure of the energy contained in the structure associated with the bump, and its position in the frequency axis will reveal its characteristic time-scale. The lowest frequency will indicate the largest time scale involved in the flow, provided that the analyzed magnitudes or the location of the probes have been correctly selected. This frequency will determine a lower bound for the value of the simulation time-averaging period.

In work titled "On the Large-Eddy Simulation modeling of wind turbine dedicated airfoils at high Reynolds numbers" [16] presented in the Turbulence, Heat, and Mass Transfer conference in 2012, this technique was used to determine the vortex shedding frequencies in a flow past an FX77-W-500 airfoil. The tests were performed at the high Reynolds number of $Re = 2.75 \times 10^6$, and at different angles of attack. As shown in Figure 2.9, the FX77-W-500 geometry presents a truncated trailing edge which triggers the formation of wake vortex regardless of the AoA. A probe was placed at location P, shown in Figure 2.9, in order to obtain information about the flow structures present on the upper airfoil surface. The flow velocities at that point contain information on the boundary layer attachment/reattachment process and the vortex shedding phenomenon since the boundary layer is stretched during the vortex formation and subsequent detachment.

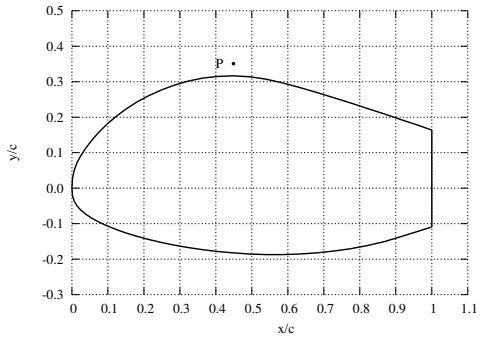

Figure 2.9: FX77-W-500 airfoil geometry. P is the location of the numerical probe in which the temporal signals of the problem variables were collected.

In the previously commented work, the power spectra methodology was used in order to study the influence of the angle of attack on the vortex shedding frequency. To do so, the angles of attack of $0°$, $12.5°$ and $16°$ were computed. In Figure 2.10, Q iso-surfaces plots are displayed for the three AoA in order to figure out the structure of the flow at each airfoil position [10]. The evolution of boundary layer detachment

process can be seen as the angle of attack rises, while the periodic shed of vortex is also clearly displayed for all AoA.

Figure 2.10: Isosurfaces of the second invariant of the velocity gradient tensor colored by the velocity magnitude on a FX77-W-500 airfoil. The flow conditons were: $Re = 2.75 \times 10^6$ at the AoA of $0°$ (top figure), $12.5°$ and $16°$ (bottom figure).

On the other hand, in Figure 2.11, the energy spectra of the streamwise velocity component obtained in the spatial location P (see Figure 2.9) are shown for the three angles of attack. All plots are included in the same chart for comparison but shifted in the vertical direction. The energy magnitudes on the vertical axis then, do not

correspond with the real values. Nonetheless, the relevant information is provided by the energy peak positions along the x axis, which reveal the frequency of the flow motions associated with it. The height of the peak also indicates the energy amount associated with the structure. Usually, the higher the peak, the lower the frequency since larger fluid structures have bigger characteristic periods (lower frequencies) and contain larger amounts of fluid mass with their associated kinetic energy.

In the specific case of Figure 2.11, it can be found a large number of peaks distributed along the power spectra, unlike in the spectrum curve that would be expected for a fully turbulent region without dominant large flow structures (see figure 2.8). All these peaks correspond to different features engendered within the flow due to complex conditions, including large pressure gradients in the leading and the trailing edges, caused by the airfoil geometry. Nevertheless, there is a peak that stands out with respect to the others at around $\kappa \approx 1$. This peak is related to the most relevant and energetic event contained in the flow, the vortex shedding process. As can be observed in the figure, the peak position along the wavenumber axis moves towards lower frequencies when increasing the AoA, from $\kappa \approx 0.806$ to $\kappa \approx 0.559$. These values give us a clear hint of the largest characteristic time scales of the flow, and they are a good indicator to estimate the total average period necessary to obtain converged statistics. This period may vary depending on the case, but must be at least one order of magnitude larger than the lowest flow frequency [17, 18, 19, 20]. In this specific case, the frequency decreases with rising AoA values, which is coherent with the fact that larger angles of attack result in larger flow blockages, entailing bigger shed vortex with longer characteristic timescales.

Other quantities can also be used in order to perform the frequency analysis. It might be of interest to use spatially integrated magnitudes gathering more extensive spatial information than a single point probe signal. For instance, it could be considered the temporal lift coefficient for a bluff body simulation. The most relevant flow events taking place near the body surface will leave a trace in the temporal coefficient signal. The lift coefficient is the non-dimensional integral value of the pressure throughout the airfoil surface (see Chapter 3 for more details), and it is affected by all the events occurring near the solid airfoil wall. Although it is true that the shape of the lift frequency spectra curve will significantly differ from the turbulence power spectrum, the energy peaks of the relevant structures and their associated frequencies will be still present.

Again, if no previous information on the flow behavior is available, the power spectra will have to be computed in running time. Thus, the duration of the simulation will have to be decided during the computation process itself since a sufficiently long temporal signal sample within the statistically stationary state is needed to obtain a correct spectrum curve.

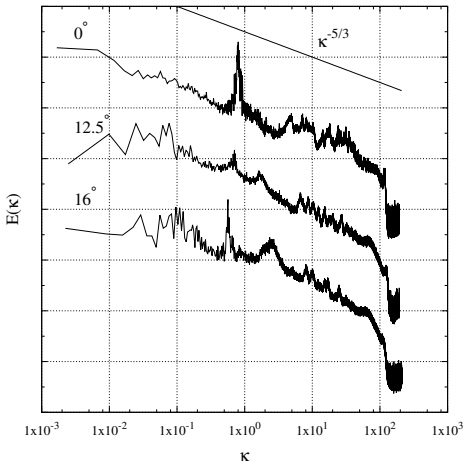

Figure 2.11: Energy spectra of the streamwise velocity component obtained in the upper surface of an FX77-W-500 airfoil at $Re = 2.75 \times 10^6$. The picture displays three different AoA, 0°, 12.5° and 16°. The plots are shifted in the vertical axis for the sake of clarity [16]

2.6 Grid resolution in the far field

As commented in Chapter 1, there are different degrees of flow modelization, from spatial and temporal fully resolving methodologies such as DNS, to highly modelized techniques, for example, the RANS approach. The size range of the resolved fluid structures may dramatically vary depending on the used methodology, and therefore, the grid resolution must be sufficient to capture the fluid motions scales required for each specific methodology. Since this Chapter is focused on time and spatial-accurate methodologies, the analysis will be performed for DNS and LES techniques. In this section, the grid requirements in the far field area (away from solid walls) will be considered, while the specific necessities of the near-wall region will be discussed in subsequent sections.

In Table 2.2, the flow resolution degree for different high fidelity CFD techniques is summarized [3]:

Table 2.2: Flow resolution requirements for four high fidelity flow modeling strategies.

Methodology	Resolution
Direct Numerical Simulation (DNS)	Turbulent motions of all scales are fully resolved
Wall Resolved Large-Eddy simulation (WRLES)	The filter and grid are sufficiently fine to resolve 80% of the energy everywhere
Wall Modeled Large-Eddy simulation (WMLES)	The filter and grid are sufficiently fine to resolve 80% of the energy away from the wall, but not in the near-wall regions
Very-Large-Eddy Simulation (VLES)	The filter and grid are too coarse to resolve 80% of the energy

As described in table 2.2, each modeling technique requires a different degree of resolved energy. Since the computational mesh is defined in terms of spatial resolution, it is necessary to relate the kinetic energy with the size of its associated flow structures. To do so, we can define a characteristic length scale (see Figure 2.12) for each different range of the energy spectrum (see Figure 2.1).

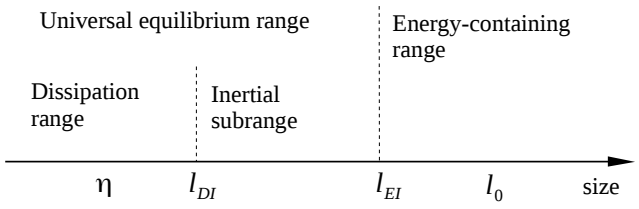

Figure 2.12: Characteristic lenghtscales associated to the different ranges of the energy spectrum. Kolmogorov scales (η), energy containing scales (l_0), limit between energy containing and inertial range (l_{EI}) and between dissipation and inertial subrange (l_{DI}).

Firstly, the Kolmogorov scale (η), is defined as the smallest structure contained in a given flow. The size of these structures is characteristic of the energy dissipation range, since the conversion process form kinetic energy to thermal energy mainly takes place at this size level. The Kolmogorov scale size η, as well as its characteristic velocity (u_η), and its characteristic time scale (τ_η) can be defined as a function of the fluid kinematic viscosity ν and the energy dissipation rate (ε):

$$\eta = (\nu^3/\varepsilon)^{1/4} \tag{2.17}$$

$$u_\eta = (\varepsilon\nu)^{1/4} \tag{2.18}$$

$$\tau_\eta = (\nu/\varepsilon)^{1/2} \tag{2.19}$$

On the other hand, we define l_0 as the characteristic length of the largest eddies, a magnitude comparable to the flow scale. The flow structures with a size of the order of l_0 belong to the energy-containing range which accounts for most of the kinetic energy contained in the fluid flow. Finally, l_{DI} and l_{EI} are the lengthscales whose associated wavenumbers ($\kappa_{DI} \sim 1/l_{DI}$ and $\kappa_{EI} \sim 1/l_{EI}$) set the bound between the inertial and dissipation spectrum ranges and the energy-containing and the inertial regions, respectively.

Pope [3] presented a set of expressions relating the flow characteristic lenghtscales (η and l_0) with the kinetic energy distribution bounds (l_{DI} and l_{EI}) such that a correspondence between the kinetic energy and the flow motion characteristic sizes is established:

$$l_{DI} = 60\eta \tag{2.20}$$

$$l_{EI} = \frac{1}{6} l_0 \tag{2.21}$$

As pointed out in Table 2.2, for a DNS, the grid resolution (i.e., the characteristic size of the discrete control volumes, h) has to be fine enough to capture the smallest dissipative scales (η). Given that this is unfeasible in most practical cases, we will place the focus on the mesh requirements for LES computations.

The LES model consists of a set of partial differential equations which is obtained after filtering the N-S equations with a filter with a characteristic cut-off length Δ. According to Langford and Moser [21], LES is a physical model since the effects of the residual motions are explicitly modeled. This physical model has to be solved through a numerical method using a mesh with a characteristic size h which has to be sufficiently small to obtain accurate solutions. According to Vreman et al. [22] and Chow and Moin [23] the appropriate ratio between the filter length Δ and the grid spacing h (h/Δ) to obtain accurate numerical solutions should be between $1/4$ and

1/2, depending on the order of the spatial numerical scheme. The higher the order, the smaller the ratio. However, in practice, when resolving the LES physical model numerically, the filter length Δ does not explicitly appear in the equations, and for high Reynolds number flows, the grid spacing is so large that h and Δ end up being the same. Nonetheless, a $h/\Delta = 1$ ratio entails some numerical error that has to be assumed.

The suitability of the grid spacing h for an accurate LES computation depends on how large this magnitude is with respect to the Kolmogorov scale size η. Pope studied the influence of the mesh grid size in the prediction accuracy of a given quantity Q through an LES model[24], assuming a constant ratio h/Δ. The study was carried out for the particular case of a free shear flow at an extremely high Reynolds number such that the Kolmogorov scale η is very small compared to the energy-containing motions of size l_0. Regarding the flow quantity Q, Q^m, is defined as the estimated value of Q modeled through the LES methodology, Q_0^m, is the value of Q obtained through a DNS, and Q_I^m, is the value of Q obtained with a grid size h within the characteristic size of the inertial subrange. The qualitative evolution of the modeled value of the flow quantity (Q^m) as a function of the grid spacing h, $Q^m(h)$, is shown in Figure 2.13 [24]. According to Pope's study, when the grid spacing size (i.e., LES filter cut-off length) is placed of the same order than η, a DNS accuracy for the Q value estimation is obtained. As h increases, the accuracy in predicting Q is progressively deteriorated. A first loss of accuracy occurs when the size of h goes beyond the dissipation range threshold (l_{DI}). Along the inertial subrange, the prediction accuracy remains constant until the inertial/energy-containing range bound is reached. Beyond this point, the accuracy is further degraded with unpredicted consequences on the numerical results.

Therefore, following Pope's analysis, the cut-off filter length, and thus, the grid spacing size h, should be set below the dissipation/inertial range limit, i.e., $h < l_{DI} = 60\eta$. Obviously, the smaller the grid size with respect to the dissipation range bound (l_{DI}), the higher the computation accuracy.

This spatial bound allowed developing a new grid quality assessment methodology. The technique is based on computing the ratio Γ between the characteristic computational cell size h and the local Kolmogorov scale η:

$$\Gamma = \frac{h}{\eta}, \tag{2.22}$$

For a well spatial-resolved simulation, Γ has to be kept below 40 units in order to ensure that the grid resolution is able to capture all the inertial range. As the ratio value lowers, the resolved part of the dissipation range will rise, transitioning from an LES to a DNS physical model when reaching values of $\Gamma \approx 1$.

According to equation 2.22, in order to obtain the ratio Γ, only two magnitudes are needed, the grid spacing h and η, being the latter a function of the turbulent kinetic

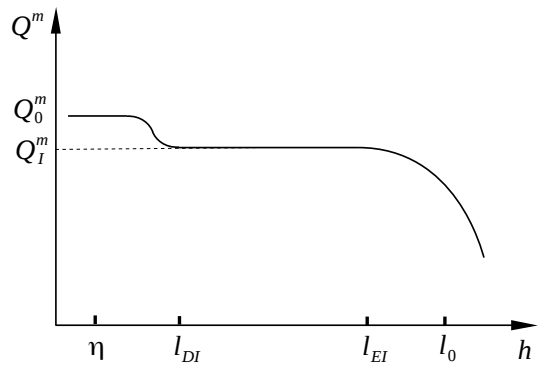

Figure 2.13: Qualitative evolution of the value of a given flow quantity Q obtained with an LES model as a function of the characteristic grid size h. Q_0^m is the value of Q obtained with a DNS, while Q_I^m, is the quantity value obtained with an LES with a grid size of the same order than the inertial subrange lenghtscale.

energy dissipation, ε, and the fluid kinematic viscosity, ν (see Equation 2.17). The value of h can be straightforwardly obtained from the mesh geometry. Since this methodology is suitable for general unstructured meshes, the grid spacing size h is evaluated according to the classical Deardorff definition, $h = v_{cell}^{1/3}$, in which v_{cell} stands for the computational cell volume.

On the other hand, to determine η, the value of ε is needed. The turbulent kinetic energy dissipation is defined as:

$$\varepsilon = 2\nu\langle S'_{ij} S'_{ij}\rangle, \tag{2.23}$$

where ν is the fluid viscosity, $\langle\cdot\rangle$ is the time-averaging operator, and S'_{ij}, is the rate-of-strain tensor of the velocity fluctuations, defined as $S'_{ij} = \frac{1}{2}\left(\partial u'_i/\partial x_j + \partial u'_j/\partial x_i\right)$.

Since the necessary data to evaluate $\langle S'_{ij} S'_{ij}\rangle$ has to be collected during simulation running time, this makes the present methodology an a posteriori verifications tool.

This strategy was used in the work "On the Large-Eddy Simulation modeling of wind turbine dedicated airfoils at high Reynolds numbers" [16]. In figure 2.14, a snapshot with the Γ distribution across the computational domain is shown for the FX77-W-500 airfoil at $Re = 2.75 \times 10^6$ and $AoA = 16°$. For this flow configuration, the airfoil features a large boundary layer detachment process as shown in figure 2.10.

Although the ratio is below the value of 40 in most domain regions, the upper area

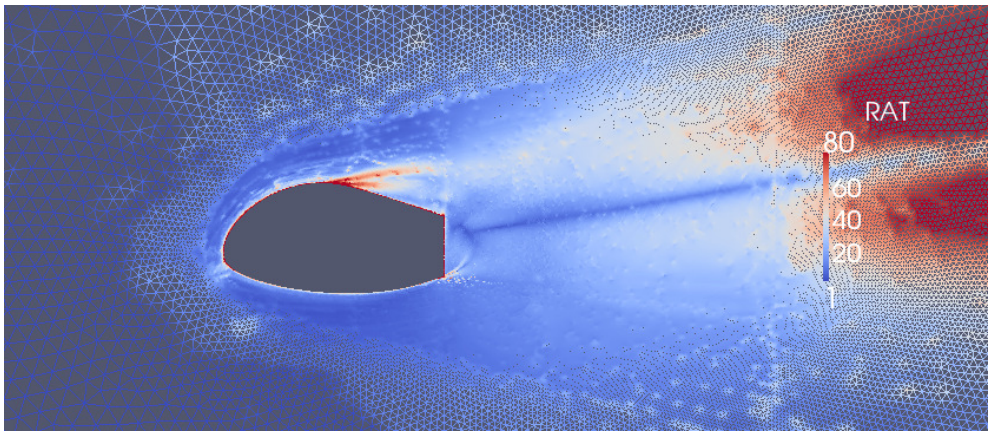

Figure 2.14: Ratio $\Gamma = h/\eta$ distribution of an LES simulation of a FX77-W-500 airfoil at $Re = 2.75 \times 10^6$ and $AoA = 16°$.

of the detached boundary layer is insufficiently resolved as the ratio value reaches values of 80. Then, a subsequent mesh refinement would be needed in that area. The near-wall region is not taken into account in this study since it needs specific considerations. In the following section, the grid requirements for that region will be detailed.

2.7 Grid resolution in the near-wall region

The near-wall mesh spacings require additional specifications compared to the far field grid resolution. Aside from resolving the flow structures present in this area, it is also required to determine the wall shear stress accurately. Thus, the wall mesh has to be fine enough to capture the steep velocity gradients in the wall-normal direction which are characteristic of that region. As commented in Chapter 1, this is a key issue to obtain reliable numerical results and has to be carefully addressed.

It is important to distinguish among the different methodologies since the degree of modelization of the wall region differs significantly, and therefore, the associated grid requirements as well. In table 2.3, the wall resolution characteristics of the different modeling methods are summarized.

Table 2.3: Summary of the grid requirements in the wall region for four different modeling approaches.

Methodology	Resolution
Direct Numerical Simulation (DNS)	Turbulent motions of all scales and the near-wall velocity profile (i.e. τ_w) are fully resolved.
Wall Resolved Large-Eddy simulation (WRLES)	The filter and grid are sufficiently fine to resolve the flow scales according to the parameters defined in Section 2.6. The near-wall velocity profile is resolved.
Wall Modeled Large-Eddy simulation (WMLES) with wall shear stress model	The filter and grid are sufficiently fine to resolve 80% of the energy away from the wall and the outer layer, but not in the inner layer. The near-wall velocity profile is not properly resolved and the τ_w must be supplied externally.
Wall Modeled Large-Eddy simulation (WMLES) with hybrid RANS/LES.	The filter and grid are sufficiently fine to resolve 80% of the energy away from the wall. The boundary layer is fully RANS modeled although the velocity profile is well resolved in the wall-normal direction.

The flow quantities scale differently depending on the wall separation. While in the inner layer, the velocity and other flow magnitudes scale with the viscous length scale (i.e., $\delta_v = \nu/u_\tau$), in the outer layer, these magnitudes scale with the boundary layer thickness. Therefore, the grid spacings should be defined accordingly depending on whether the inner layer is resolved or not [25, 26]. Thus, since in DNS and WRLES the inner layer is resolved, the grid spacings will be defined in wall (viscous) units, while in WMLES, a fraction of the boundary layer thickness will be used.

2.7.1 DNS requirements

The grid requirements for DNS have been mainly determined through numerical studies of canonical flows such as the Turbulent Boundary Layer, the Pipe or the Channel flow. The latter has been widely studied with a broad range of Reynolds numbers allowing a deep understanding of turbulent boundary layer physics[14, 27, 28, 29, 30, 31, 32].

In the near-wall region, the important large-scale structures are the "streaks" [33], which extend along the streamwise direction and are scarcely spaced in the spanwise one (see Figure 1.1, $y^+ \approx 4$). Their mean spacing characterizes the lengthscale of the viscous sublayer structures including the sublayer thickness. Therefore, an insufficient mesh resolution in the spanwise direction will probably entail a thicker viscous sublayer compared to its physical counterpart. This may lead to lower velocity gradients at the wall (i.e., lower wall shear stress) and will trigger all the consequences derived from an excessively low τ_w [27].

According to the studies carried out by Kim, Moin and Moser [28], the near-wall streaks in a turbulent channel flow are highly anisotropic. In the streamwise direction, their characteristic size is approximately $l_x^+ \approx 500$ wall units, while in the span-wise direction, the streaks are approximately $l_z^+ \approx 100$ viscous units.

These characteristic sizes may give us a rough idea of the difficulties that arise when dealing with industrial relevant flows. It can be estimated that there are of the order of 10^8 streaks on the wings of a Boeing 777 during cruise. A DNS would require to fully resolve all 10^8 streaks, which is completely unaffordable with the current technology [24].

Regarding the wall-normal direction, the first off-wall node has to be placed well within the viscous sublayer ($y^+ < 5$) to adequately capture the 3D structure of the streaks as well as the appropriate wall velocity gradient. The mean velocity profile in the viscous sublayer is linear, allowing a precise evaluation of the wall-normal velocity gradient and hence, the wall shear stress.

Since several points are needed to capture a given flow structure, the typical near-wall grid spacings used in DNS are: $\Delta x^+ \approx 10$, $\Delta z^+ \approx 7$ and $\Delta y_w^+ << 1$ [14, 28].

On the other hand, the wall-parallel grid spacings are kept constant away from the wall, while values up to $\Delta y^+ \approx 5$ are used in the center of the channel.

2.7.2 WRLES requirements

Although the grid requirements are not as severe as in DNS, in WRLES they are still quite demanding. Chapman[34] and Kravchenko et al. [35] suggested somewhat larger grid spacings compared to the DNS standards based on different LES experiences in Channel flows. However, capturing the near-wall streaks is imperative if no specific model for the wall is used, making the WRLES approach still an in-

tractable model for most practical flows. Given the restrictions commented above, Chapman suggested the values of $\Delta x^+ \approx 100$ and $\Delta z^+ \approx 20$. However, in practical WRLES applications, the suitable grid spacings values in the near-wall region are: $\Delta x^+ \approx 50 \sim 130$, $\Delta z^+ \approx 15 \sim 30$ and $\Delta y^+ < 1$ [35, 36]. Additional grid spacing requirements were also suggested in the wall-normal direction for the boundary layer viscous region (i.e., $l_y^+ < 100$). Chapman suggested placing at least 10 grid points between the solid wall and the limit of the viscous region, while Kravchenko et al. recommended between $10 \sim 30$ points.

On the other hand, Larsson also proposed grid spacings values for the outer layer region [26]. Since the flow structures present in this region scale with the local boundary layer thickness δ rather than the viscous length (δ_v), the length scales are defined in terms of δ [25, 26]. Specifically, the average sizes of the turbulent eddies in that region are $\Delta x = \Delta z \approx \delta/10$[34], while in the log layer, the size of the flow structures is proportional to the wall distance [3].

Considering the turbulent structure sizes commented above, the recommended grid spacings in the outer layer are: $\Delta x \approx 0.08\delta$ and $\Delta z \approx 0.05\delta$. Regarding the wall-normal direction, it is recommeded to set $\Delta y \approx 0.02\delta$ at the beginning of the outer layer (i.e. $y \approx 0.2\delta$) growing linearly to $\Delta y \approx 0.05\delta$ at the edge of the boundary layer ($y \approx \delta$) .

2.7.3 WMLES requirements for wall shear stress methods

In WMLES, there is a significant difference in mesh requirements with respect to the wall-resolving methodologies commented above. In this approach, the inner viscous layer is not resolved but modeled, and therefore, only the lengthscales of the outer layer flow structures have to be represented. Since the viscous sublayer structures are not resolved, an external model has to be used to supply an accurate wall shear stress to the numerical simulation.

Accordingly, since the outer layer is fully resolved in an LES sense, the grid spacings proposed by Larsson for this region in WRLES also apply for the present methodology[26]. On the basis of several WMLES studies from different authors [37, 38], the proposed values are: $\Delta x \approx 0.08\delta$, $\Delta z \approx 0.05\delta$ and $\Delta y_w \approx 0.02\delta$.

2.7.4 WMLES requirements for hybrid RANS/LES

In the hybrid RANS/LES approach, the whole boundary layer is modeled through a RANS model and therefore, no turbulent structures are resolved, mitigating the grid spacing requirements compared to the WSSM methods. For these methodologies, there is no upper limit for the wall-parallel grid spacings (i.e., Δx and Δz), allowing a reduction in computational costs up to 10 to 100 times compared to WSSM methods

[26]. On the other hand, in the wall-normal direction, the grid spacing requirement is $\Delta y_w^+ < 1$ in order to properly capture the near-wall mean velocity profile, which is essential to evaluate a correct velocity gradient and therefore, the wall shear stress. For a wider discussion on grid requirements for hybrid RANS/LES methods, the reader is referred to the following references [39, 40].

2.7.5 Summary

In table 2.4, a summary of the grid spacings for both, the inner and the outer layer are given for different high fidelity modeling strategies. The grid parameters are in terms of boundary layer thickness and viscous units depending on the region. Hence, previous knowledge of the flow is required since both, δ and τ_w are needed to obtain the dimensional value necessary to generate the computational mesh. If no experimental o numerical data is available, an initial numerical simulation would be required in order to estimate the boundary layer parameters.

Table 2.4: Summary of the required grid spacings for four different high fidelity modeling strategies. The grid parameters are given for the inner layer (in case of a wall-resolving approach) and the outer layer.

Methodology	**inner layer**			**outer layer**		
	Δx^+	Δy^+	Δz^+	Δx	Δy	Δz
DNS	≈ 10	$<< 1$	≈ 7	$\approx 10^+$	$< 1^+$ / $< 5^+$	$\approx 7^+$
WRLES	$50 \sim 130$	< 1	$15 \sim 30$	0.08δ	$> 0.02\delta$ / $< 0.05\delta$	0.05δ
WSSM WMLES	N/A	N/A	N/A	0.08δ	$\Delta y_w \approx 0.02\delta$	0.05δ
RANS/LES WMLES	N/A	< 1	N/A	N/A	N/A	N/A

2.8 Conclusions

In this Chapter, a set of methodologies intended to ensure the quality of high fidelity numerical simulations has been described. Accurate numerical simulations resolve the fluid flow from the temporal and the spatial point of view, being necessary to ensure that both, the spatial domain size and the temporal simulation span are sufficient to contain all the relevant flow features. The two-point correlation function and the energy spectrum have been proposed to evaluate the domain size and the simulation period suitability, respectively. In general, these parameters are dependent on the largest flow scales, which are explicitly resolved in all high fidelity approaches.

Therefore, the evaluation methodologies and requirements for the domain size and the time-averaging span assessment, are common for all modeling methods.

On the other hand, regarding the numerical discretization, the spatial and temporal resolutions have to be appropriately set depending on the simulation methodology and its particular requirements which have been detailed in the present Chapter. These parameters are related to the small-scale flow structures, and therefore, they vary according to the degree of modelization. The smaller the resolved flow structures, the higher the simulation fidelity and the required grid and temporal resolutions. In this regard, it is important to highlight the significant differences in grid requirements between the far field and the near-wall areas. Boundary layers are multiscale phenomena in which even the smallest flow structures are essential to determine the flow development correctly. This poses a major difficulty for high fidelity simulations of wall-bounded flows, becoming necessary the use of external models to evaluate the wall shear stress which is a key magnitude to obtain reliable numerical results.

Regarding the far-field resolution assessment, the Γ methodology, which evaluates the ratio between the smallest flow scales and the local grid size, was developed during the present Ph.D. The methodology was initially tested with several flows past wind turbine dedicated airfoils [16].

As it has been pointed out all along the Chapter, the major drawback of the assessment methodologies is that all of them rely on previously known numerical or experimental data of the flow that is being simulated. Any evaluation methodology would need this information since, except the smallest turbulent flow structures in the dissipation range, the rest are far from being universal and strongly depend on the particular flow conditions. In general, there is no an a priori knowledge of the flow configuration, and the only way to obtain an estimation of the flow evolution is performing an a priori simulation. Unfortunately, there is no a universal recipe that would allow to correctly set a given simulation without having previous knowledge of the flow.

Bibliography

[1] R. Courant, K. Friedrichs, and H. Lewy. Über die partiellen Differenzengleichungen der mathematischen physik. *Mathematische Annalen.*, 100:32–74, 1928.

[2] C.B. Laney. *Computational gasdynamics.* Cambridge university press., 1998.

[3] S. B. Pope. *Turbulent flows.* Cambridge Univesity Press, 2000.

[4] V. Kolár. Vortex identification: New requirements and limitations. *Int. J. Heat Fluid Fl.*, 28, 2007.

[5] H.J Lugt. Vortex flow in nature and technology. *Wiley Panton R.L.*, 28, 1984.

[6] M.S. Chong, A.E. Perry, and B.J. Cantwell. A general classification of three-dimensional flow field. *Phys. Fluids A*, 2:765–777, 1990.

[7] J. Jeong and F. Hussain. On the identification of a vortex. *J. Fluid Mech.*, 285:69–94, 1995.

[8] S. Kida and H. Miura. Identification and analysis of vortical structures. *Eur. J. Mech. B/Fluids*, 17:471–488, 1998.

[9] R. Cucitore, M. Quadrio, and A. Baron. On the effectiveness and limitations of local criteria for the identification of a vortex. *Eur. J. Mech. B/Fluids*, 18:261–282, 1999.

[10] J.C.R. Hunt, A.A. Wray, and P. Moin. Eddies, streams, and convergence zones in turbulent flows. *Center for Turbulence Research, Stanford, CA*, N89-24555:193–208, 1988.

[11] G. Comte-Bellot. Contribution a l'etude de la turbulence de counduite. *Doctoral thesis.*, 1963.

[12] C.B. da Silva and J.C.F. Pereira. Invariants of the velocity-gradient, rate-of-strain, and rate-of-rotation tensors across the turbulent/nonturbulent interface in jets. *Phys. Fluids*, 20:055101, 2008.

[13] A. Pont-Vílchez, F.X. Trias, A. Gorobets, and A. Oliva. Direct numerical simulation of backward-facing step flow at $\mathrm{Re}_\tau = 395$ and expansion ratio 2. *Under review.*, pages 25–37, 2018.

[14] A. Lozano-Durán and J. Jiménez. Effect of the computational domain on direct simulations of turbulent channels up to $\mathrm{Re}_\tau = 4200$. *Phys. Fluids*, 26:011702–1, 2014.

[15] J. Calafell, F.X. Trias, O. Lehmkuhl, and A. Oliva. A time-average filtering technique to improve the efficiency of Two-Layer Wall Models for Large Eddy Simulation in complex geometries. *Comput. Fluids*, 2019.

[16] J. Calafell, O. Lehmkuhl, I. Rodríguez, and A. Oliva. On the large-eddy simulation modeling of wind turbine dedicated airfoils at high Reynolds numbers. In *Turbulence, Heat and Mass Transfer 7, Palermo, Italy*. Begell House, Inc., 2012.

[17] I. Rodríguez, R. Borrell, O. Lehmkuhl, C.D. Perez-Segarra, and A. Oliva. Direct numerical simulation of the flow over a sphere at Re = 3700. *J. Fluid Mech.*, 679:263–287, 2011.

[18] I. Rodríguez, O. Lehmkuhl, R. Borrell, and A. Oliva. Direct numerical simulation of a NACA0012 in full stall. *Int. J. Heat Fluid Flow*, 43:194–203, 2013.

[19] O. Lehmkuhl, I. Rodríguez, R. Borrell, J. Chiva, and A. Oliva. Unsteady forces on a circular cylinder at critical Reynolds numbers. *Phys. Fluids*, 26:125110–1–125110–22, 2014.

[20] O. Lehmkuhl, I. Rodríguez, J. Chiva, R. Borrell, and A. Oliva. On the flow past a circular cylinder from critical to super-critical Reynolds numbers: Wake topology and vortex shedding. *Int. J. Heat Fluid Flow*, 55:91–103, 2015.

[21] J. Langford and R. Moser. Optimal les formulations for isotropic turbulence. journal of fluid mechanics. *J. Fluid Mech.*, 398:321–346, 1999.

[22] B. Vreman, B. Geurts, and H. Kuerten. Large-eddy simulation of the turbulent mixing layer. *J. Fluid Mech.*, 339:357–390, 1997.

[23] F.K. Chow and P.A. Moin. A further study of numerical errors in large-eddy simulations. *J. Comput. Phys.*, 184(2):366–380, 2003.

[24] S. Pope. Ten questions concerning the large-eddy simulations of turbulent flows. *New J. Phys.*, 6(35):1–24, 2004.

[25] G. I. Park and P. Moin. Numerical aspects and implementation of a two-layer zonal wall model for LES of compressible turbulent flows on unstructured meshes. *J. Comput. Phys.*, 305:589–603, 2016.

[26] J. Larsson, S. Kawai, J. Bodart, and I. Bermejo-Moreno. Large eddy simulation with modeled wall-stress: recent progress and future directions. *B. JSME*, 3:1–23, 2016.

[27] P. Moin and J. Kim. Numerical investigation of turbulent channel flow. *J. Fluid Mech.*, 118:341–377, 1982.

[28] J. Kim, P. Moin, and R. Moser. Turbulence statistics in fully developed channel flow at low Reynolds number. *J. Fluid Mech.*, 177:133–166, 1987.

[29] J. Jimenez and P. Moin. The minimal flow unit in near-wall turbulence. *J. Fluid Mech.*, 225:213–240, 1991.

[30] S. Hoyas and J. Jiménez. Scaling of the velocity fluctuations in turbulent channels up to $\mathrm{Re}_\tau = 2003$. *Phys. Fluids*, 18:011702, 2006.

[31] O. Flores and J. Jiménez. Hierarchy of minimal flow units in the logarithmic layer. *Phys. Fluids*, 22:071704, 2010.

[32] A. Lozano-Durán, O. Flores, and J. Jiménez. The three-dimensional structure of momentum transfer in turbulent channels. *J. Fluid Mech.*, 694:100–130, 2012.

[33] S.J. Kline, W.C. Reynolds, F.A. Schraub, and P.W.. Runstadler. The structure of turbulent boundary layers. *J. Fluid Mech.*, 30:741–773, 1967.

[34] D. R. Chapman. Computational aerodynamics, development and outlook. *AIAA J.*, 17:1293–1313, 1979.

[35] A.G. Kravchenko, P. Moin, and R. Moser. Zonal embedded grids for numerical simulations of wall-bounded turbulent flows. *J. Comput. Phys.*, 127:412–423, 1996.

[36] H. Choi and P. Moin. Grid-point requirements for large eddy simulation: Chapman's estimates revisited. *Phys. Fluids*, 24:011702, 2012.

[37] S. Kawai and J. Larsson. Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy. *Phys. Fluids*, 24:015105, 2012.

[38] J. Lee, C. Minjeong, and H. Choi. Large eddy simulations of turbulent channel and boundary layer flows at high Reynolds number with mean wall shear stress boundary condition. *Phys. Fluids*, 25:110808, 2013.

[39] P. R. Spalart, S. Deck, M. L. Shur, K. D. Squires, M. Kh. Strelets, and A. Travin. A new version of detached-eddy simulation, resistant to ambigous grid densities. *Theor. Comput. Fluid Dyn.*, 20:181–195, 2006.

[40] P. R. Spalart. Detached-eddy simulation. *Annu. Rev. Fluid Mech.*, 41:181–202, 2009.

3

Two-Layer wall model for non-equilibrium flows

3.1 Introduction

In the introduction Chapter, the huge difficulties of fully resolving the boundary layer characteristic lengthscales, especially at high Reynolds numbers, have been widely discussed. Different modeling strategies intended to overcome or at least mitigate the unaffordable computational costs derived form the near-wall flow resolution have been reviewed, while the strengths and weaknesses of each particular methodology have been analyzed. The present work is focused on wall shear stress models for LES. This approach may entail significant savings in terms of spatial and temporal resolution while keeping a moderate degree of modelization. Nonetheless, the potential gains of this method are quite dependent on the boundary layer characteristics. From an LES grid resolution point of view, the number of required mesh points is strongly dependent on the boundary layer thickness. The thicker the boundary layer, the lower the required mesh resolution [1]. On the other hand, the physical complexity of the near-wall flow may also strongly affect the advantages of using a wall shear stress model. In general, the required degree of physical and mathematical generality grows with the flow complexity. Prior to apply a given wall shear stress model to a specific flow, is must be ensured that the physical formulation of the selected model is able to reproduce the phenomena involved in the flow that is being numerically reproduced. The boundary layer complexity may range from relatively simple equilibrium wall flows, whose boundary layers feature a rather universal behavior [2, 3, 4], to very specific and complicated flow regimes affected by strong adverse pressure gradients, flow detachments or recirculations, and other conditioning factors that may affect their behavior. Unfortunately, in general, the higher the governing equations physical

complexity, the higher their resolution costs, which may undermine the advantages of using a given wall model.

The most relevant findings of the present section have been published in the journal article entitled "A time-average filtering technique to improve the efficiency of Two-Layer Wall Models for Large Eddy Simulation in complex geometries." in Computers & Fluids journal [5].

3.1.1 Two-layer models

This Chapter is focused on the zonal or two-layer model (TLM) approach proposed by Balaras *et al.* [6, 7] in the mid-'90s. This methodology was briefly introduced in the introduction Chapter (1). The approach is based on the resolution of the RANS equations or a simplified variant in a fine auxiliary mesh that is embedded in the LES domain between the solid boundary and the first off-wall node (see Figure 3.1). Depending on the mathematical and physical formulation of the model, its range of applicability may vary from simple equilibrium flows to very complex non-equlibrium conditions. The model governing equations are resolved by taking boundary conditions from the LES fluid nodes while the time integration is carried out with implicit or semi-implicit schemes to avoid restrictions related to stability considerations.

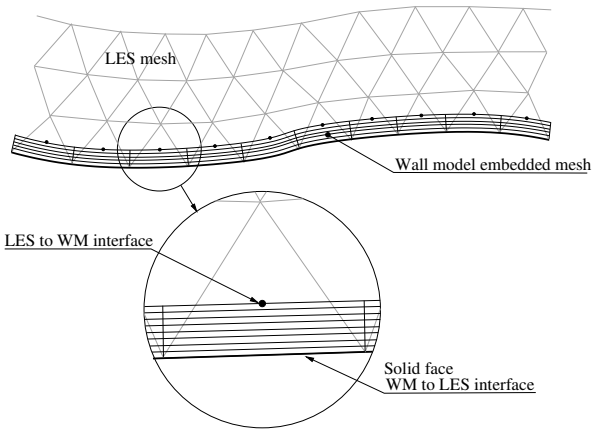

Figure 3.1: The two-layer strategy scheme.

The resolution of a set of governing equations allows having an accurate near-wall velocity field from which an accurate wall shear stress can be derived. This magnitude is the model output that is fed back into the LES domain as a boundary condition.

Two-Layer wall models have been recurrently studied since they represent a good physical model for Large Eddy Simulations with underresolved wall regions. Specifically, those based on the Reynolds Averaged Navier-Stokes equations are of special interest, since they can be applied to a wide range of conditions including non-equilibrium flows. Nonetheless, these models are affected by two recurrent problems, the "log-layer mismatch" (LLM) and the resolved Reynolds stresses (RSS) inflow, which until now, have been dealt with separated techniques[8, 9]. In this work, a time-filtering methodology is proposed to tackle both issues at once with a single and low-computational-cost step, easily applicable to complex three-dimensional geometries. The time-filtering technique has already been applied to other types of wall models to mitigate the "log-layer mismatch" [10]. Nonetheless, in the present work, it is applied for the first time in the Two-Layer wall model context, showing its ability not only in avoiding the mismatch issue but also in blocking the resolved Reynolds stress inflow. This approach dramatically improves the wall model performance and generality compared to other existing implementations[8, 9] which were reviewed in the introduction section. Additionally, a methodology to determine the necessary temporal filter length is proposed and validated in equilibrium and non-equilibrium conditions. On the other hand, the filter size influence on large-scale unsteady flow motions is assessed. Good results are obtained in steady and unsteady flow regimes by suppressing the LES highest frequencies while taking into account large-scale temporal effects. Hereafter, the two characteristic problems of the LLM and RRS inflow are described in more detail.

The resolved Reynolds stresses inflow

Resolved turbulent motions and their associated fluctuating velocity field have inherent diffusive effects on the mean flow [11]. In the Navier-Stokes physical model, these effects are not explicitly introduced but they are an indirect consequence of the time-resolved fluctuations. Following Pope's nomenclature, this source of diffusivity will be named as apparent diffusivity. On the other hand, when turbulent scales are not fully resolved, their diffusive effects can be introduced through a model. This source of diffusivity is named as modeled diffusivity, and unlike the apparent one, it is explicitly introduced through an extra viscosity added to the diffusive term. The RANS approach is intended to exclusively resolve the mean flow, while the effects of the whole fluctuating component (Reynolds stresses) are modeled through the turbulent viscosity, ν_T. Therefore, if additional time-resolved fluctuations are introduced in a RANS domain through the boundary by the advective term, their apparent diffusive effects will be added to the RANS modeled ones, which already account for the whole fluctuating field. This causes the skin friction overprediction pointed out by Cabot *et al.* [12, 13]. This effect is called resolved Reynolds stresses inflow.

Kawai and Larsson proposed a new approach to deal with the RRS inflow problem.

They showed that κ coefficient in the wall layer turbulence model could not remain constant in the wall-normal direction given that the ratio between resolved/modeled Reynolds stressed varies dramatically in that direction [14, 15, 16]. They proposed to modify the value of κ to account for the apparent diffusive effects of the RRS. On the other hand, Park and Moin [9] proposed to explicity quantify these effects through a viscous-like magnitude, ν_{Pa}, which is explicitly subtracted to the RANS turbulent eddy viscosity. This approach helps in keeping appropriate diffusivity levels within the wall layer. However, the method requires a spatial average of the variables along a homogeneous direction to obtain the resolved Reynolds stress tensor component, which severely restricts the generality of the model regarding the allowed geometries [9, 17]. In the introduction section, the RRS inflow resolution methodologies are discussed in more detail.

The "log-layer mismatch"

Some of the previous works pointed out the impossibility of having accurate LES velocity data at the first few off-wall nodes, mainly due to the use of inadequate LES models and large numerical errors in the near-wall region when using coarse meshes [13]. These errors were supposed to cause the so-called "log-layer mismatch" (LLM) issue, an unphysical rise of the fluid velocity in the boundary layer logarithmic region [8]. By contrast, Yang *et al.* [10] argued that when using data from the first off-wall row of nodes, the wall shear stress and the LES velocity fluctuations are artificially in phase. This instantaneous synchronization is unphysical and causes unreal damping of the Reynolds shear stresses ($-\langle u'v' \rangle$) near the wall. This effect is compensated by a rise of the velocity gradient ($\langle \partial u/\partial y \rangle$) to fulfill the momentum balance in the first off-wall nodes of a channel flow with a constant pressure gradient ($-\langle u'v' \rangle + \langle \left(\nu + \nu_{sgs}\right) \partial u/\partial y \rangle) = (1/\rho)\partial p/\partial x = ct$).

Kawai and Larsson [8, 14] proposed to stretch the wall layer mesh beyond the first off-wall node, reducing numerical and subgrid modeling errors caused by the wall proximity and suppressing the LLM issue. They obtained satisfactory results in the LES domain for the region above the WM/LES interface. On the other hand, Yang *et al.* [10] proposed to apply a time-averaging filter (TAF) at the WM/LES interface to deal with the LLM problem. They validated the approach through a simple logarithmic-law-like equilibrium wall model. The effects of the TAF were only tested in a high Reynolds number Channel Flow. Nonetheless, good results were obtained even taking LES data from the first off-wall node. According to Yang *et al.*, the reason why the application of a TAF eliminates the LLM, is that the temporal coupling between the first off-wall LES velocity and the wall shear stress is broken, eliminating the unphysical damping of the Reynolds shear stresses. This also explains the ability of Kawai's methodology in preventing the LLM since taking the LES data away from the wall, also eliminates the artificial synchronization.

The present work aims at going further into the generalization of the TLM approach to allow its application in a range of flows as wide as possible, especially to those with practical industrial applications such as high Reynolds number aerodynamics in complex geometries. To do so, an improved and efficient two-layer wall model based on the full incompressible unsteady Reynolds-averaged Navier-Stokes (URANS) equations is proposed for unstructured meshes. The model features a time-averaging filter in the WM/LES interface. As previously commented, this approach was firstly introduced by Yang *et al.* for a simple equilibrium wall function, obtaining promising results in suppressing the LLM issue. In this work, the TAF technique is applied for the first time to a TLM. In this context, the mission of the TAF is not only suppressing the LLM, but also to block the RRS inflow. It is the first time that a TAF is used for this purpose. Moreover, a methodology to determine the optimal temporal filter length for a RANS-based TLM is proposed.

Firstly, in Section 3.2, the LES and WM mathematical and numerical strategies are detailed. Then, in Section 3.3, the ability of the TAF in filtering the incoming RRS from the LES solution is evaluated through a numerical test based on WRLES Pipe Flow at $Re_\tau \approx 500$. Additionally, a systematic methodology to determine the optimal value of the filter length is porposed and discussed.

Then, in Section 3.4, the model is validated in equilibrium and non-equilibrium conditions with a canonical turbulent Pipe Flow at $Re_\tau \approx 3000$ (Section 3.4.1) and a stalled DU91-W2-250 airfoil at $Re = 3 \times 10^6$ (Section 3.4.2), respectively. The effects of the filter length on the numerical predictions are analyzed, including its influence on transient phenomena such as the airfoil boundary layer detachment/reattachment process.

After validating the technique intended to obtain the optimal filter length, the wall model is tested and validated for fully 3D geometries, specifically, an Ahmed car body at $Re = 7.68 \times 10^5$. In all tests, numerical results are compared with direct numerical simulation (DNS) and/or experimental data depending on the case.

On the other hand, the interaction of the wall model with the LES model is discussed. The Pipe Flow at $Re_\tau \approx 3000$ test is performed with 5 different subgrid models with very different wall treatments in order to assess the behavior of the global WM/LES model.

Finally, conclusions are given in Section 3.6.

3.2 Mathematical and numerical model

In this section, the governing equations and their numerical resolution methods are presented for the LES and the wall model domains.

3.2.1 The LES mathematical formulation and numerical resolution

In the LES approach, the filtered Navier-Stokes (N-S) equations are numerically solved in order to obtain a direct solution for the scales larger than the filter size. The smaller subgrid scales are not solved, thus their important physical effects are taken into account through the subgrid stress tensor. The LES equations are obtained by spatially filtering the incompressible N-S equations:

$$\nabla \cdot \overline{\mathbf{u}} = 0, \tag{3.1}$$

$$\frac{\partial \overline{\mathbf{u}}}{\partial t} + (\overline{\mathbf{u}} \cdot \nabla)\,\overline{\mathbf{u}} + \nabla \overline{p} - \nu \nabla^2 \overline{\mathbf{u}} = \nabla \cdot \left(\overline{\mathbf{u}}\,\overline{\mathbf{u}}^T - \overline{\mathbf{u}\mathbf{u}^{\mathbf{T}}} \right) \approx -\nabla \cdot \tau(\overline{\mathbf{u}}), \tag{3.2}$$

where $\overline{(\cdot)}$ is a spatial commutative filtering operator, $\mathbf{u}$ the velocity field, p the kinematic pressure and $\tau(\overline{\mathbf{u}})$ the subgrid stress tensor which is modeled according to the Boussinesq hypothesis for incompressible flows:

$$\tau\left(\overline{\mathbf{u}}\right) = -2\nu_{sgs} S(\overline{\mathbf{u}}), \tag{3.3}$$

where $S(\overline{\mathbf{u}})$ is the rate-of-strain tensor, $S(\overline{\mathbf{u}}) = \frac{1}{2}\left(\nabla \overline{\mathbf{u}} + \nabla \overline{\mathbf{u}}^T\right)$, and ν_{sgs} is the subgrid viscosity. Notice that $\tau\left(\overline{\mathbf{u}}\right)$ is considered traceless without the loss of generality, because the trace can be included as part of the filtered pressure, $\overline{p}$.

To close the formulation, a suitable expression for ν_{sgs} must be provided. Most of the published works related to TLM [7, 8, 9, 13], used the dynamic Smagorinsky model of Germano *et al.* [18] or its variant for compressible flow of Moin *et al.* [19] with the modifications of Lilly [20]. Therefore, in the present work, the same model will be used to allow comparison with other TLM formulations in the literature.

On the other hand, the dynamic Smagorinsky of Germano *et al.* was not used in the Ahmed car body test of Aljure *et al.* [21]. In order to allow comparison with their results, in the Ahmed car test, the Wall-Adaptive Large Eddy-Viscosity (WALE) model of Nicoud[22] has been applied.

Regarding the numerical resolution of the filtered N-S equations, it is carried out through the finite volume method. The equations are discretized on a collocated unstructured grid arrangement by using second-order symmetry-preserving schemes [23, 24]. These schemes are conservative, i.e., they preserve the symmetry properties of the continuous differential operators and ensure both, stability and conservation of the kinetic energy balance even at high Reynolds numbers and using coarse grids [25]. On the other hand, the temporal discretization of the momentum equation has been done through a second-order one-step explicit scheme for the convective and diffusive terms[26], while for the pressure, an implicit first-order scheme has been used. Finally, a fractional-step method is applied to solve the pressure-velocity coupling.

3.2.2 The wall model mathematical and numerical formulation

In order to model the near-wall flow field, the URANS equations are numerically solved:

$$\frac{\partial \mathbf{U}}{\partial t} + (\mathbf{U} \cdot \nabla)\,\mathbf{U} = \nabla \cdot [2(\nu + \nu_{Twm})S(\mathbf{U})] - \nabla P, \tag{3.4}$$

Here, the capital letters mean time-averaged variables and ν_{Twm} is the RANS turbulent viscosity for the WM. The resolution is performed in a fine embedded mesh that stretches from the solid wall to a given height. For the validation tests, the extrusion length will be set to match the first off-wall row of nodes height (Δy_1), although it can be a user-defined parameter (see Figure 3.1). The URANS equations have been chosen in order to have the lowest possible degree of approximation by taking into account all the non-equilibrium terms.

To complete the model, a RANS model must be selected to evaluate ν_{Twm}. A mixing-length-based algebraic model together with the Van Driest wall-damping function is used like in most of TLM implementations found in the literature[7, 9]:

$$\nu_{Twm} = (\kappa y)^2 \,|S|\, \left[1 - exp\left(-y^+/A^+\right)\right]^2, \tag{3.5}$$

where $\kappa = 0.41$ is the von Kármán constant, y is the wall distance, $|S|$ is the magnitude of the rate-of-strain tensor and $A^+ = 26$, is a constant of the wall-damping function. The superindex $+$ denotes a magnitude in wall units. Even though the law of the wall is implicit in this approach and therefore not valid for non-equilibrium flows, successful results have been reported in several works [9, 15] when using it.

In the introduction, the LLM and the RRS inflow problems were discussed, including their resolution methodologies proposed so far. As previously commented, in the present work, a TAF is applied in the WM/LES interface to tackle both issues at once, with a single and low-computational-cost methodology.

The average filtering of the LES variables is carried out through a numerical exponential running average method[9, 27], which is defined as follows for a given variable ϕ:

$$\overline{\phi}^n = (1-\epsilon)\overline{\phi}^{n-1} + \epsilon\phi^n \quad ; \quad \epsilon = \frac{\Delta t/T}{1 + \Delta t/T}, \tag{3.6}$$

where $\overline{(\cdot)}$ is the time filtering operator, Δt is the time-step at iteration n, and T is the filter characteristic time-scale or filter length. The filtering operation in equation 3.6 is the numerical solution to the ordinary differential equation $\partial\overline{\phi}/\partial t = (\phi - \overline{\phi})/T$, whose exact solution is $\overline{\phi}(t) = \int_0^t \phi(\xi)\frac{exp[(\xi-t)/T]}{T}d\xi$ [9]. Therefore, $\overline{\phi}$ may be considered as the local time-average of ϕ with an exponential decaying memory, being the decaying speed dependent on the value of T.

The optimal value of T is widely discussed in subsequent sections. Specifically, in Section 3.3, the minimum necessary filter length allowing to avoid the RRS inflow is determined through a numerical experiment. Then, in the validation Section 3.4, the influence of T on the numerical results of both, mean and unsteady magnitudes is evaluated.

Regarding the numerical resolution of the model Equations (3.4), the finite volume method is applied. Second-order symmetry-preserving numerical schemes have been used to carry out the spatial discretization of the convective term, while a second-order central difference scheme has been applied for the diffusive one. The velocity-pressure coupling has been solved through a projection method while the temporal integration is carried out by means of the Euler first-order implicit scheme. The present implicit projection method includes a pressure-correction step to allow second-order accuracy in the pressure field resolution[28]. The implicit approach is used to avoid numerical stability issues given that the time-step used for the WM computation, is the CFL of the LES domain but applied to a much finer mesh.

On the other hand, Dirichlet boundary conditions are prescribed at the top boundary (see Figure 3.2) for velocities and pressure taking their values from the time-filtered LES variables, whereas, at the solid faces, no-slip and Neumann conditions are applied for velocities and pressure, respectively. Finally, in the case of existing side boundaries in the WM mesh, appropriate boundary conditions must be set depending on the characteristics of each particular case. The side boundaries are generated by extruding the edges of an open surface, and when they are coincident with an LES mesh boundary, the same boundary condition is applied to both surfaces.

In general, it is difficult to control the LES node positions when dealing with unstructured meshes, and a mismatch between the LES and WM nodes at the interface surface could easily occur. To minimize numerical inaccuracies, a second-order approximation is used to interpolate the LES values to the WM nodes on the wall model top boundary following the methodology proposed by Park and Moin [29]. These interpolations can cause a small mass imbalance which is corrected at each LES iteration to ensure a zero net mass flow through the external boundaries of the WM mesh.

Finally, once the near-wall velocity profile is obtained, an accurate wall shear stress is derived from the expression $\tau_i \approx \mu\, u_{iy_1} / \delta_{y_1}$, and then, it is supplied to the LES computation to evaluate the diffusive term at the solid face. In the previous expression, μ is the dynamic viscosity while u_{iy_1}, is the i^{th} velocity component at the first off-wall node with respect to a wall-parallel coordinate system ij and δ_{y_1} is the distance between this node and the wall (see Figure 3.2).

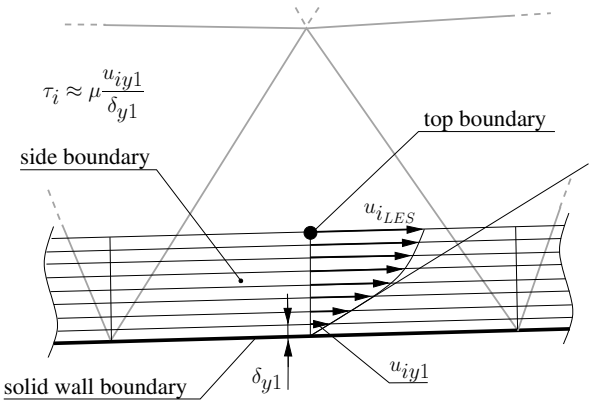

Figure 3.2: Wall model grid scheme.

3.3 Time-averaging filter performance test

In this section, the ability of the TAF in blocking the RRS inflow into the wall layer domain is assessed through a numerical experiment.

The strategy consists in applying the TLM to an independent Pipe Flow WRLES computation. The WM is fed with time-resolved data taken from the LES domain at the WM/LES interface. This data is used as boundary condition to solve the WM equations but the LES domain is not fed back with the TLM output. Then, the input time-resolved signal is successively filtered with increasing averaging periods, T, to determine the filter length influence on the RRS levels within the WM domain, and the consequences of this on the wall layer physical predictions.

Specifically, the numerical setup is based on a wall-resolved pipe flow at $Re_\tau = u_\tau R/\nu \approx 500$, where R is the pipe radius. Furthermore, the TLM is embedded between the wall and a height of $y^+ \approx 150$, well into the pipe logarithmic law region, which is a plausible operation position of the WM at high Reynolds number. On the other hand, the tested filter lengths are determined by analyzing the power spectrum of the streamwise velocity component at the WM/LES interface. This technique reveals the characteristic time-scales of the flow structures present in that region. These turbulent structures generate the RRS, which are transported into the wall layer through the WM/LES interface causing the wall shear stress overprediction. Since the underlaying behavior of the energy spectrum of a turbulent flow is universal, this methodology could provide a rather general criterion to establish the necessary filter length for a given flow.

To quantify the presence of RRS in the wall model domain and their apparent diffusive effects, the technique originally proposed by Park and Moin[9] to deal with the diffusivity excess in the TLM context is now used for this other purpose. As commented in the introduction section, Park and Moin modeled the apparent diffusivity inherent to the RRS through a viscosity-like magnitude, ν_{Pa}, which was subtracted to the RANS ν_T to keep appropriate diffusivity levels within the wall layer. Given the physical meaning of ν_{Pa}, its non-dimensional form, ν_{Pa}/ν, is a suitable quantity to measure the apparent diffusive effects of RRS in the WM domain.

According to Park and Moin, the viscous-like quantity, ν_{Pa}, can be obtained with the following expression:

$$\nu_{Pa} = -\frac{R(\mathbf{U})S^d(\mathbf{U})}{2S^d(\mathbf{U})S^d(\mathbf{U})}, \tag{3.7}$$

where $S^d(\mathbf{U})$, is the deviatoric part of the rate-of-strain tensor, and $R(\mathbf{U})$, is the resolved part of the Reynolds stress tensor. Although the evaluation of $R(\mathbf{U})$ requires the spatial average in an homogeneous direction[9, 17], Equation 3.7 is applicable for the present test.

On the other hand, the ability of the model in reproducing the flow physics, is evaluated through the mean streamwise velocity profile and the wall shear stress computed by the WM. DNS data of Chin *et al.*[30] is used as a reference for the mean velocity profiles, while the computed Re_τ, is used as a measure of the wall shear stress and compared to the reference value of $Re_\tau \approx 500$.

3.3.1 Test setup

The WRLES computations are performed in a domain of length $8R$. This is well above the minimum length of $2\pi R$ required to accurately resolve the one-point first and second-order statistics[30] at $Re_\tau \approx 500$. The computational mesh has been generated by extruding a plane mesh along the streamwise direction. The two-dimensional (2D) mesh is a structured mesh with square-shaped control volumes (CV) in the region between $r = 0.5R$ and $r = R$ while between the pipe center and $r = 0.5R$ an all-triangles unstructured pattern is applied. This mesh arrangement in the pipe core is intended to avoid the wedge-shaped cells at the pipe axis which cause a significant unphysical time-step reduction. The total number of grid points of the LES mesh is 6×10^6 distributed as follows: in the outer region ($r \in [0.5R, 1.0R]$), $N_z = 256$, $N_\theta = 192$ and $N_r = 60$, being z the streamwise, θ the azimuthal and r the radial directions while in the inner region ($r \in [0, 0.5R]$), the unstructured mesh is also extruded in 256 planes. The grid spacings in wall units are $\Delta z^+ \approx 15$, $\Delta r\theta^+ \approx 16.5$ at $r = R$ and $\Delta r^+ \approx 1.2$ at the wall, being $\Delta y_1^+ \approx 0.6$. Concerning the WM setup, the WM/LES interface is placed at $y^+ \approx 150$, matching the top boundary nodes with their

LES counterparts to avoid interpolation inaccuracies. The WM mesh is extruded in 20 layers and refined towards the wall, being the first off-wall node well into the viscous sublayer at a distance of $y_1^+ = 0.54$ and making a total amount of 9.8×10^5 inner nodes. Respecting the subgrid strategy for the LES domain, the dynamic Smagorinsky model has been used. Periodic boundary conditions are prescribed in the streamwise direction while no-slip and Neumann conditions are applied to the wall for velocity and pressure, respectively. The flow is enforced by keeping a constant mass flow consistent with the Reynolds number of $Re = 2R\overline{U}/\nu$ which is based on the bulk velocity $\overline{U}$. In this case $Re = 1.7 \times 10^4$ which corresponds to a $Re_\tau \approx 500$ according to the Blasius correlation.

To gain insight on the flow structures present in the WM/LES interface and their characteristic time-scales, the power spectrum of the streamwise velocity component is computed in the WM/LES linking region (see Figure 3.3). This will provide valuable data about the temporal behavior of the resolved Reynolds stresses that are being introduced into the wall layer.

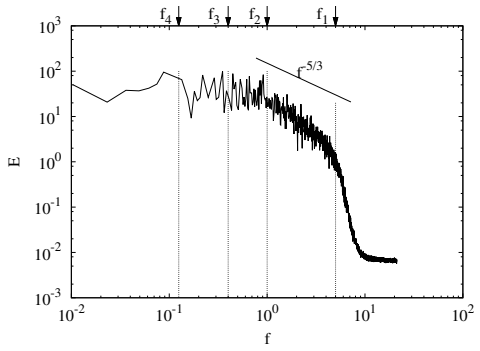

Figure 3.3: Power spectra of the streamwise velocity component at $y^+ \approx 150$ of a Pipe flow at $Re_\tau \approx 500$.

The characteristic flow frequencies revealed by the power spectrum will be used to determine the TAF filter lengths to be tested. Four frequencies have been selected in the plot. Firstly, two in the energy-containing range in which the largest flow structures are located, $f_1 = 0.125$ and $f_2 = 0.4$. The first one corresponds to the lowest frequency that can be captured by the present domain length. Then, other characteristic frequencies are considered by choosing the limit between the energy-containing and the inertial subrange, $f_3 = 1.0$, and the limit between the latter and the dissipation range, $f_4 = 5.0$. Therefore, the first test will be performed without filtering the LES input data. Then, successive filter lengths of $T_1 = 0.2$, $T_2 = 1.0$, $T_3 = 2.5$ and

$T_4 = 8.0$, corresponding to the frequencies commented above ($T_n = 1/f_n$), will be applied in the WM/LES interface. Both, the averaging periods and the frequencies are defined in terms of the non-dimensional time units (TU) and its inverse, respectively. In Table 3.1, the different filter configurations are summarized.

Table 3.1: Summary of the different filter configurations used for the TAF performance test. The tested filter lengths and their position in the energy spectrum are shown.

TAF Config.(n)	Filter length T_n	$f_n = 1/T_n$	Energy spectrum range
0	no filter	no filter	N/A
1	0.2	5.0	inertial/dissipation range limit
2	1.0	1.0	inertial/energy-containing range limit
3	2.5	0.4	within the energy-containing range
4	8.0	0.125	flow-through period, largest flow scales

Regarding the temporal integration and statistics averaging periods, the WRLES computation has been advanced during 100 flow-through cycles until reaching the statistically stationary regime. Once the steady state has been achieved, the WM has been coupled to the LES solution, and an additional transient period of 100 TU has been run. Finally, the wall layer averaged variables have been collected along 100 TU which is more than 10 times the largest flow structure appearing in the velocity spectrum.

3.3.2 Time-averaging filter performance test results

In Table 3.2, results concerning the Re_τ computed by the WM are given for each TAF configuration in Table 3.1. Each case is identified with a symbol to easily locate the test results in Figures 3.4 and 3.5.

Table 3.2: TAF performance test results performed with a channel flow at $Re_\tau \approx 500$. The WM computed Re_τ and its relative error with respect to the target value ($Re_\tau \approx 500$) is shown. Symbols are to identify the numerical results in Figures 3.4 and 3.5.

Test (n)	Symbol	Filter length T_n	Computed Re_τ	rel. err. [%]
0	○	no filter	528.70	5.74
1	△	0.2	515.66	3.13
2	▽	1.0	506.81	1.36
3	□	2.5	502.06	0.41
4	◇	8.0	502.18	0.43

On the other hand, in Figure 3.4, the mean velocity profiles for all TAF configurations are shown and compared with the DNS data of Chin *et al.*[30].

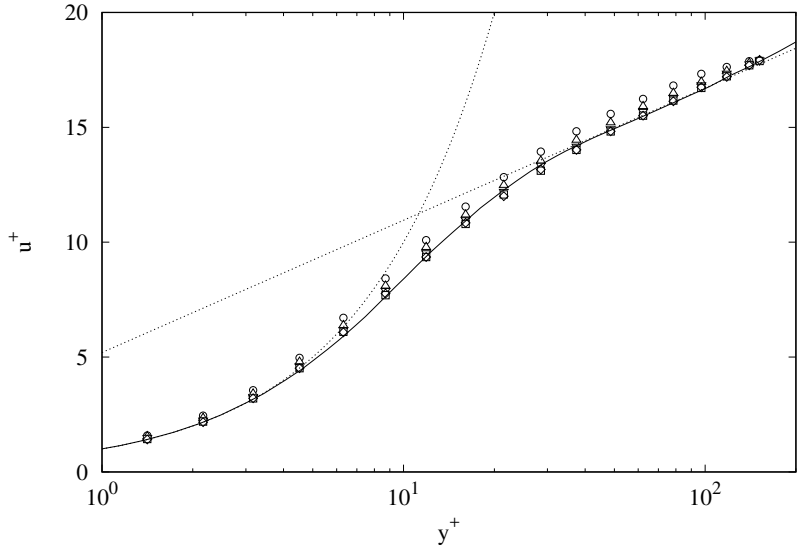

Figure 3.4: Mean streamwise velocity profiles of the wall layer region of a pipe flow at $Re_\tau \approx 500$. Comparison between DNS data (————) and the WM mean velocity field for five different TAF configurations (dots). ○ no TAF, △ $T_1 = 0.2\,TU$, ▽ $T_2 = 1.0\,TU$, □ $T_3 = 2.5\,TU$ and ◇ $T_4 = 8.0\,TU$. (See Table 3.2). The law of the wall (·····).

The tests confirm the findings of Cabot and Moin [13] about the incoming RRS. The excess of diffusivity due to the combination of a RANS model with the turbulent resolved inflow causes an overprediction of the wall shear stress, as the computed Re_τ value indicates in case 0 in which no measure was taken to avoid that. As the filter length is increased, the shear stress converges progressively to the reference Re_τ value. According to the values displayed in Table 3.2, satisfactory results are obtained when the cut-off filter length is set approximately at frequencies lower than inertial subrange ones. For higher frequencies (at the beginning of the dissipation range) the averaging period is not enough to sufficiently block the RRS inflow, although a significant improvement is observed compared to the non-filtered solution. Similar conclusions can be reached by inspecting the wall layer mean velocity profiles in

Figure 3.4. Good convergence with the DNS solution is obtained when frequencies from the beginning of the inertial subrange upwards are suppressed.

To analyze in deeper detail the effects of RRS inflow on the wall layer physics prediction, the ν_{Pa}/ν quantity is displayed in Figure 3.5 (left). The application of the TAF significantly reduces the apparent diffusive effects of the RRS in the wall domain, even with the smallest filter length. This is consistent with the fact that the smallest filtering period, $T_1 = 0.2$, suppresses the LES dissipation range frequencies. Nonetheless, for the smallest filter cut-off length, the total diffusivity levels are still too high. This unphysical extra diffusivity is indirectly caused by the remaining RRS through the WM RANS model. This can be observed in Figure 3.5 on the right, in which the non-dimensional turbulent viscosity, ν_{Twm}/ν, is shown. According to the mixing-length-based RANS model (see Equation 3.5), the excessive levels of ν_{Twm} are due to an overpredicted value of the rate-of-strain tensor magnitude, $|S|$. This is a consequence of the incoming time-resolved velocity field of the inertial subrange, given that the RANS model expects only mean-flow-based quantities. On the other hand, the Van Driest wall-damping function could not cause the unphysical ν_{Twm} values, since erroneous values of y^+ derived from a wall shear stress overprediction, would shrink the function damping region towards the wall.

In summary, the highest frequencies are the main responsible for the apparent dissipative effects of the RRS. However, in the RANS context, filtering the smallest flow-scales and their dissipative component is not enough. The inertial subrange motions have an unphysical contribution to the rate-of-strain magnitude which leads to an incorrect evaluation of ν_T. For a RANS model, $|S|$ should be exclusively derived from the mean velocity field, or from very-large-scale unsteady motions at most[31]. Regarding the latter, according to the results, the lowest frequencies of the energy-containing range do not significantly affect the RANS model. For the largest filter lengths, $T_3 = 2.5$ and $T_4 = 8.0$, the results remain unchanged. As a conclusion, the minimum filter cut-off length has to be set at the energy-containing/inertial range limit.

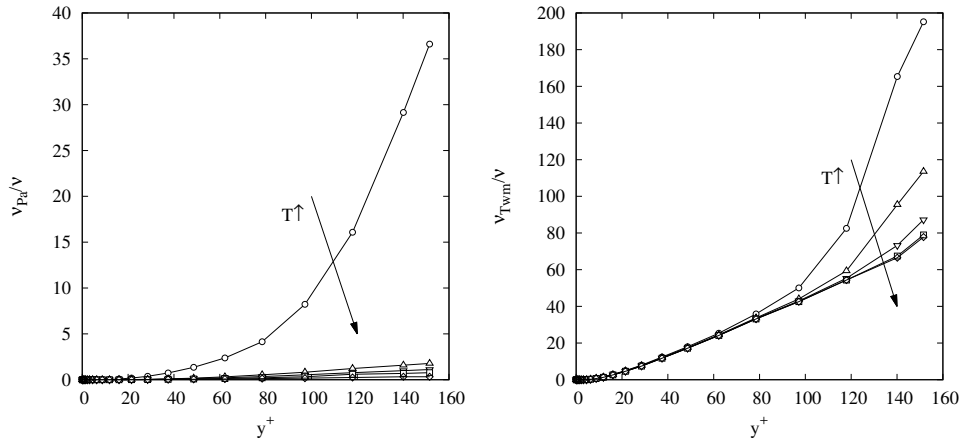

Figure 3.5: Left: Normalized viscosity obtained with the Park and Moin methodology as a measure of the the RRS levels ($\nu_{Pa}/\nu = -(R(\mathbf{U})S(\mathbf{U})/2\nu S(\mathbf{U})S(\mathbf{U})$ Right: Normalized turbulent viscosity (ν_{Twm}/ν). Both magnitudes are plotted vs. the wall distance in wall units and within the wall layer. See Table 3.2 for symbols definition.

3.4 Model validation

In this section, the proposed methodology is validated with two different cases, an equilibrium turbulent Pipe Flow at $Re_\tau \approx 3000$, and a non-equilibrium flow around a DU 91-W2-250 airfoil, at high Reynolds number and stall condition. Specifically, the flow conditions for the airfoil are, $Re = u_\infty c/\nu = 3 \times 10^6$, based on the airfoil chord (c) and the free stream velocity (u_∞), and an angle of attack (AoA) of 15.2°. The effects of the time averaging period on the numerical results will be analyzed by testing different T values, following the same methodology used in the previous section. The consequences of the filter length on unsteady flow conditions will be also studied through the analysis of boundary layer detachment/reattachment frequencies in the stalled airfoil test.

The WMLES results will be compared with DNS data from Ahn *et al.*[32] for the Pipe Flow test, while experimental data from Timmer and van Rooij[33] will be used in the airfoil test for validation.

3.4.1 Pipe flow at $Re_\tau \approx 3000$

Test description

The DNS data of Ahn *et al.*[32] used for reference, corresponds to a Pipe Flow test at $Re_\tau = 3008$, and was obtained in a computational domain of length $30R$ with periodic conditions in the streamwise direction.

In our LES and WMLES computations, a 10R-long pipe has been used due to the large number of WM configurations to be tested. Despite that this domain length is insufficient to capture the very large structures that are engendered in Pipe and Channel flows at high Reynolds numbers[32], Lozano-Durán and Jiménez[34] showed that one-point statistics, which are the results analyzed below, remain unaffected with reasonably small domains. The authors[34] performed tests with a Channel flow at $Re_\tau = u_\tau \delta / \nu = 4200$ and a domain size of $L_x \times L_y \times L_z = 2\pi\delta \times 2\delta \times \pi\delta$, being x the streamwise, y the wall-normal, and z the spanwise directions, while δ is the channel half height. The channel length was about one third shorter than the present pipe flow simulation. Moreover, according to Ahn *et al.*[32], the effects of very large scales are more severe in Channel than in Pipe flows. This allows us to conclude that the length of $10R$ is large enough to evaluate the mean velocity profiles and the velocity fluctuations accurately.

The same procedure as in Section 3.3 has been followed to generate the computational grid. The total number of points of the LES mesh is 8.9×10^5 distributed as follows: a structured pattern for $r \in [0.5R, 1.0R]$ with $N_z = 128$, $N_\theta = 96$ and $N_r = 25$, while an all-triangle unstructured mesh is used in the pipe core. The grid spacings in wall units are $\Delta z^+ \approx 236$, $\Delta r\theta^+ \approx 198$ at $r = R$ and $\Delta r^+ \approx 60$, being the first off-wall LES nodes placed at $\Delta y_1^+ \approx 30$, at the beginning of the logarithmic region.

The tested TAF configurations are summarized in Table 3.3. The filter lengths, T, have been chosen through the energy spectrum of the streamwise velocity component obtained at the WM/LES interface ($y^+ \approx 210$), according to the methodology proposed in Section 3.3.

Table 3.3: Summary of the different TAF configurations tested with the $Re_\tau \approx 3000$ Pipe Flow.

TAF Config.(n)	Filter length T_n	$f_n = 1/T_n$	Energy spectrum range
0	no filter	no filter	N/A
1	0.125	8.0	dissipation range
2	0.55	1.8	inertial/dissipation range limit
3	2.0	0.5	inertial/energy-containing range limit
4	4.0	0.25	within the energy-containing range
5	10.0	0.1	flow-through period, largest flow scales

This numerical experiment, aside from evaluating the effects of the filter length T on the overall LES results, it is also intended to assess the ability of the TAF methodology in dealing with the LLM and the RRS inflow problems separately. To do so, the technique proposed by Kawai and Larsson [8] of extending the WM mesh to suppress the LLM will be used to uncouple the effects of the TAF on each specific problem. Therefore, the TAF configurations detailed in Table 3.3, will be applied at two different WM extrusion heights, specifically, at $h_{wm}^{+} \approx 30$ and at $h_{wm}^{+} \approx 210$, which corresponds to the first and fourth off-wall node rows, respectively.

Thus, the WM grid resolution in the wall-normal direction is 10 layers when using an extrusion height of $h_{wm}^{+} \approx 30$, and 20 in case of $h_{wm}^{+} \approx 210$. The nodes are conveniently concentrated towards the wall according to a hyperbolic tangent law, being the position of the first off-wall nodes $y^{+} \approx 0.1$ in both cases, i.e., well into the viscous sublayer. The wall model cell count is 1.2×10^5 and 2.4×10^5, respectively.

The boundary conditions and the flow enforcement methodology are the same as in Section 3.3. In this case, the Reynolds number based on the bulk velocity is $Re = 1.33 \times 10^5$, which corresponds to $Re_\tau \approx 3026$ according to the Blasius correlation. Finally, a period of 100 flow-through cycles is computed to allow the flow to reach a statistically steady turbulent state, while the averaged variables are collected over 150 additional flow-through loops.

Test results

For convenience, from here onwards in this section, the notation followed by Ahn *et al.*[32] regarding the coordinate axis and velocity components will be used. The cylindrical coordinates are converted to Cartesian ones, being the new streamwise component $x = z$, the wall-normal direction $y = 1 - r$ and the spanwise direction $z = r\theta$. Regarding the velocity components, $u = u_z$, $v = -u_r$ and $w = u_\theta$.

In Table 3.4, the computed Re_τ is shown for the six different TAF configurations specified in Table 3.3 at the two different extrusion heights. On the other hand, in order to assess the overall improvement caused by the WM in the LES solution, a LES-only computation, which uses the same LES mesh as the WMLES but without WM, is also displayed. In brackets, the relative error of the computed Reynolds number with respect to the reference value of $Re_\tau \approx 3026$ is given.

Table 3.4: Computed Re_τ values of the pipe flow at $Re = 1.33 \times 10^5$ which corresponds to $Re_\tau \approx 3026$. The values are obtained for six different TAF filter lengths summarized in Table 3.3. Each TAF configuration is tested with a WM/LES interface height of $h^+_{wm} \approx 30$ and $h^+_{wm} \approx 210$. The LES-only solution is also displayed. The relative error in % with respect the reference value, $Re_\tau \approx 3026$, is in brackets.

TAF Config.(n)	Filter length T_n	No WM	WM $h^+_{wm} \approx 30$	WM $h^+_{wm} \approx 210$
0	No Filter	1923.6(36.4)	3409.2 (12.66)	3316.1 (9.58)
1	0.125	N/A	3305.6 (9.23)	3256.9 (7.63)
2	0.55	N/A	3201.1 (5.78)	3161.9 (4.49)
3	2.0	N/A	3141.3 (3.81)	3116.7 (3.0)
4	4.0	N/A	3138.0 (3.70)	3105.5 (2.62)
5	10.0	N/A	3135.6 (3.62)	3100.4 (2.45)

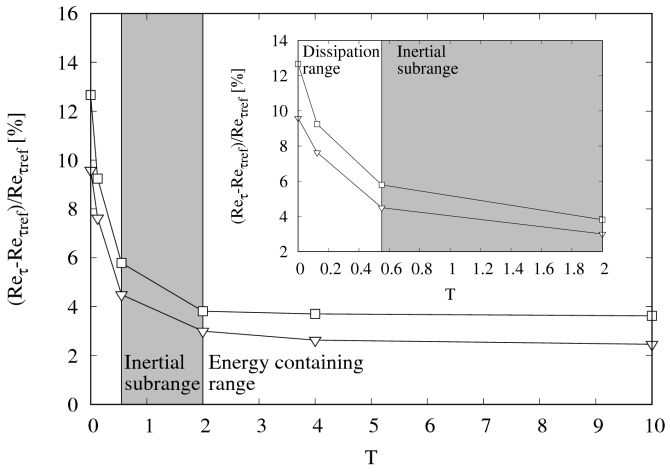

Figure 3.6: Relative error of the computed Re_τ in % with respect the reference value of $Re_\tau \approx 3026$ vs. the time-averaging period T. The values are obtained for the WM configurations in Table 3.4. $h^+_{wm} \approx 30$ (□) and $h^+_{wm} \approx 210$ (▽). On the top-right corner, the dissipation and the inertial subrange regions are zoomed.

In the same way as in the TAF performance test, the error in the wall shear stress prediction is minimized when frequencies in the inertial subrange or higher are filtered. For larger filtering lengths, the error keeps almost independent of T. On the other hand, the effects of the TAF on the LLM can be indirectly evaluated by analyzing

the results obtained at the two different extrusion heights. When no TAF is applied, in the $h_{wm}^{+} \approx 210$ test, the RRS inflow problem is isolated since the LLM error is avoided according to Kawai and Larsson's methodology [8]. By contrast, in the $h_{wm}^{+} \approx 30$ test, both problems coexist. In the latter case, the error in the wall shear stress evaluation is approximately 3% higher than for the larger extrusion height. When applying the TAF, a significant drop of the Re_τ error is observed for both tests. Nonetheless, the error difference narrows to only 1% for larger filter lengths, suggesting that in the $h_{wm}^{+} \approx 30$ case, the TAF is not only acting on the RRS inflow, but also on the LLM problem. It is worth to notice that the Kawai and Larsson's technique is slightly more efficient in tackling the LLM problem than Yang's one. However, the error difference is around only 1%, which according to our opinion, it does not justify the huge computational costs and geometrical difficulties associated to the use of Kawai and Larsson's approach.

In Figures 3.7, 3.8 and 3.9, plots of the mean streamwise velocity profile of the LES domain are displayed in wall units and logarithmic scale. The information is distributed as follows: in Figure 3.7, the gains of using an extended WM mesh are assessed by plotting WMLES results obtained for both extrusion heights, $h_{wm}^{+} \approx 30$ and $h_{wm}^{+} \approx 210$. To isolate the effects of the WM mesh height, the value of T has been set at $T_3 = 2.0$, in the inertial/energy-containing range limit, at which the effects of the RRS inflow are sufficiently minimized. Additionally, LES-only data is also displayed to evaluate the overall improvement obtained by using the WM in the optimal configuration.

On the other hand, Figures 3.8 and 3.9 are intended to analyze the influence of T on the mean velocity profile. The analysis is carried out separately for each extrusion heights, $h_{wm}^{+} \approx 30$ in Figure 3.8, and $h_{wm}^{+} \approx 210$ in Figure 3.9. The selected filter lengths correspond to the TAF configurations shown in Table 3.3 (from 0 to 3). For the sake of clarity, the largest averaging periods (configurations 4 and 5) are not displayed since they present minor differences with respect to configuration 3. In all plots, the DNS solution of Ahn *et al.*[32] is displayed for comparison.

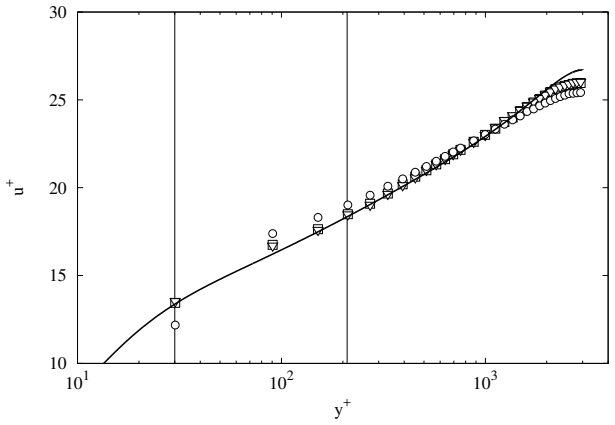

Figure 3.7: Mean streamwise velocity profiles in wall units . The wall distance is in logarithmic scale. The results have been obtained with the following configurations: LES-only (○), WMLES at $h_{wm}^{+} \approx 30$ (□) and WMLES at $h_{wm}^{+} \approx 210$ (▽). Both WMLES solutions are obtained with $T_3 = 2.0$. The vertical lines represent the two LES/WM interface heights. DNS Ahn *et al.*[32] (————).

According to data in Figure 3.7, with frequencies in the dissipation and inertial range suppressed, the WM is able to reproduce the characteristic logarithmic region for both extrusion heights. Although the higher extrusion height of $h_{wm}^{+} \approx 210$ seems to perform slightly better than the lower configuration at $h_{wm}^{+} \approx 30$, only minor differences are observed. On the other hand, in Figures 3.8 and 3.9, progressive improvements are obtained with rising values of T for both extrusion heights. Nonetheless, the gains are more significant in the $h_{wm}^{+} \approx 30$ configuration. In this particular case, the non-filtered solution is simultaneously affected by the LLM and the RRS inflow problems, increasing the overall prediction error which is globally mitigated by the TAF. By contrast, the velocity profile obtained with $h_{wm}^{+} \approx 210$, is only affected by the RRS inflow. In fact, the non-filtered solution obtained with Kawai and Larsson's method is reasonably good except for the first off-wall node, so that the impact of the TAF on the LES results was more limited than in the $h_{wm}^{+} \approx 30$ configuration.

As previously commented, the mean flow profile as well as the wall shear stress, remain constant for values of T larger than 2.0. This suggests that having a correct mean momentum balance is the primary requirement to obtain correct first-order statistics while having a realistic time-resolved wall shear stress is of relative importance. These results are in line with those obtained by Yang *et al.* [10], and they are

consistent with the findings of Piomelli and Balaras [35], who showed that for poorly resolved wall regions, only the average effects of the near-wall structures must be represented by the wall layer model.

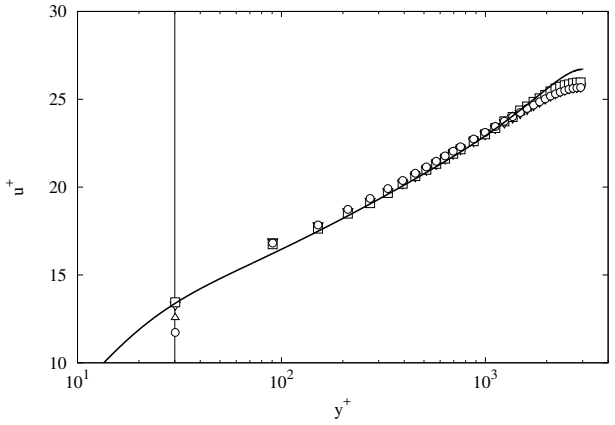

Figure 3.8: Mean streamwise velocity profiles in wall units. The wall distance is in logarithmic scale. The results have been obtained in the following conditions: WMLES at $h_{wm}^+ \approx 30$ without TAF (○) and with TAF with filtering periods of $T_1 = 0.125$ (△), $T_2 = 0.55$ (▽) and $T_3 = 2.0$ (□). See Table 3.3 for details. The vertical line represents the LES/WM interface position. DNS Ahn *et al.*[32] (————).

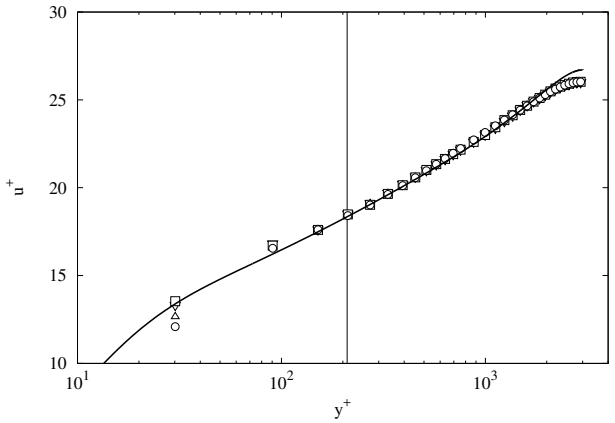

Figure 3.9: Mean streamwise velocity profiles in wall units. The wall distance is in logarithmic scale. The results have been obtained in the following conditions: WMLES at $h_{wm}^+ \approx 210$ without TAF (○) and with TAF with filtering periods of $T_1 = 0.125$ (△), $T_2 = 0.55$ (▽) and $T_3 = 2.0$ (□). See Table 3.3 for details. The vertical line represents the LES/WM interface position. DNS Ahn *et al.*[32] (————).

Finally, in Figure 3.32, the root-mean-square of the velocity fluctuations (u'_{rms}, v'_{rms}, and w'_{rms}) as well as the Reynolds shear stress, $\langle u'v' \rangle$, is shown for the two extrusion heights, $h_{wm}^+ \approx 30$ (left) and $h_{wm}^+ \approx 210$ (right), with the same increasing values of T used for the mean streamwise velocity. Again, for a sufficiently large averaging period ($T \gtrsim 2.0$), the WMLES results converge to a very similar solution regardless of the WM height. The converged solutions are in fair agreement with the DNS reference except in the near-wall region, especially for the streamwise and the spanwise velocity fluctuations. This is due to the necessity of having a minimum amount of resolved scales to capture second-order statistics properly. Since the size of the eddies in the near-wall region is proportional to the wall distance and, therefore, very small compared with those in the pipe center, the mesh is unable to capture a minimum amount of near-wall structures when using coarse meshes with a constant grid spacing. According to the results, it appears that the WM is not able to correct this. On the other hand, the velocity fluctuation results obtained with time averaging periods of 4.0 and 10.0 were very similar to those computed with $T_3 = 2.0$. This suggests that the fluctuation levels are more related to a well-predicted mean flow and a sufficiently fine mesh than having an accurate temporal resolution of the wall shear stress.

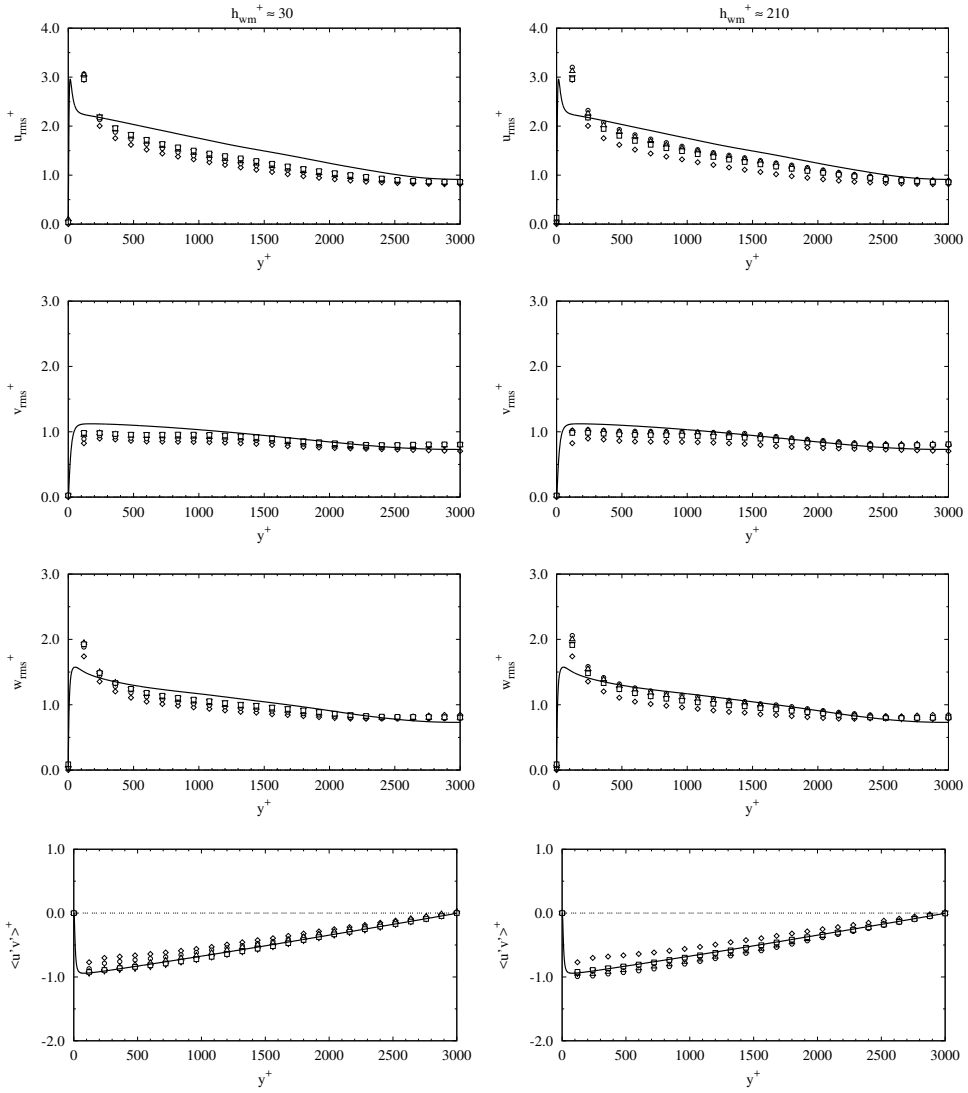

Figure 3.10: rms of the streamwise (u'^{+}_{rms}), wall-normal (v'^{+}_{rms}) and azimuthal (w'^{+}_{rms}) velocity fluctuations as well as the Reynolds shear stress $(\langle u'v' \rangle^{+})$ in wall units. The results have been obtained in the following conditions: LES-only ($\diamond$) and WMLES at $h^{+}_{wm} \approx 30$ (left) and $h^{+}_{wm} \approx 210$ (right). In case of WMLES, the wall model configurations are: no TAF ($\circ$), and TAF with filtering periods of $T_1 = 0.125$ ($\triangle$), $T_2 = 0.55$ ($\triangledown$) and $T_3 = 2.0$ ($\square$). See Table 3.3 for details. DNS Ahn *et al.*[32] (————).

3.4.2 Flow around a DU 91-W2-250 airfoil at Reynolds number $Re = 3 \times 10^6$ and $AoA = 15.2^\circ$

Test description

The flow around a DU 91-W2-250 airfoil at high Reynolds number $(Re = 3 \times 10^6)$ and in full stall $(AoA = 15.2^\circ)$, is studied to evaluate the WM performance in non-equilibrium conditions as well as to analyze the effects of the TAF averaging period on unsteady phenomena. In this particular case, the effects of T on the boundary layer detachment/reattachment frequency will be analyzed.

The DU 91-W2-250 profile is a wind turbine dedicated airfoil which is designed to have a premature laminar to turbulent transition to avoid performance degradation caused by dirtiness accumulation. It has been reported [9] that in the laminar portion of the boundary layer, the WM turbulence model must be switched off to obtain laminar profiles and, therefore, a more appropriate wall shear stress. However, a methodology to detect the laminar region has to be implemented, and this is out of the scope of the present work. For that reason, a profile with almost no laminar boundary layer section has been selected. On the other hand, wind tunnel data from Timmer and van Rooj[33] is available for comparison. It includes the integral lift and drag coefficients as well as the pressure coefficient distribution.

The computational domain of the simulation is a square region of $40c \times 40c \times 0.3c$, where x, y and z axis are the chord-wise, chord-normal and spanwise directions, respectively (see Figure 3.11). The leading edge of the airfoil is placed at the square center.

The airfoil simulation methodology has been taken from previous experiences in which numerical results and experimental data were compared obtaining good agreement between them [36]. To set an appropriate spanwise distance, L_z, the conclusions of the LESFOIL project [37] have been followed. In that project, LES simulations of an Aerospatiale A-airfoil at similar flow conditions (i.e., $AoA = 13.3^\circ$ with flow separation and $Re = 2.1 \times 10^6$) were carried out. The authors concluded that a spanwise distance of approximately 1.5 times the flow separation distance in the wall-normal direction at the trailing-edge, was acceptable. According to preliminary computations which were subsequently confirmed, in the present case, the separation distance is approximately $0.2c$. Thus, a L_z of $0.3c$ has been set and further validated through two-point correlation in the spanwise direction. To do so, two numerical probes have been placed at maximum turbulent kinetic energy positions on the airfoil surface [38] and within the recirculation region, specifically at P1 $(0.61, 0.17)$ and P2 $(0.89, 0.20)$ (see Figure 3.11 right). The two-point correlations have been computed according to Equation 3.8 for the fluctuations of the three velocity components and pressure:

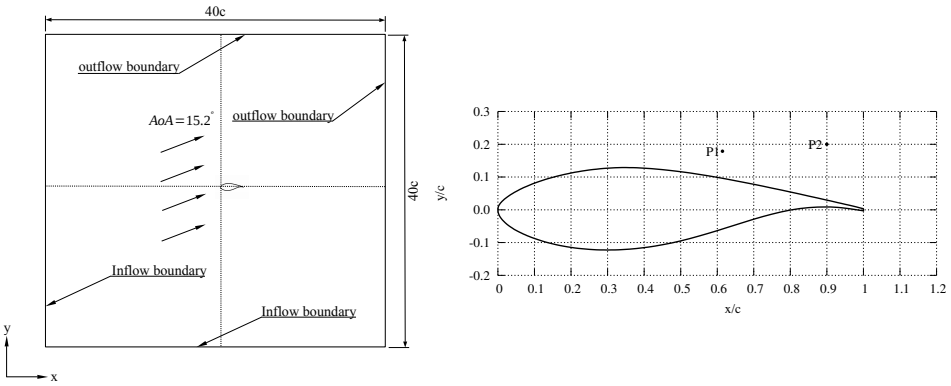

Figure 3.11: Left: Computational domain of the DU 91-W2-250 simulation at $Re = 3 \times 10^6$ and $AoA = 15.2^\circ$. Right: DU 91-W2-250 geometry and position of probes P1 and P2.

$$\Re_{\phi\phi}(\mathbf{x}, \delta z) = \frac{\langle \phi'(\mathbf{x}, t)\phi'(\mathbf{x} + \delta z, t)\rangle}{\langle \phi'(\mathbf{x}, t)^2\rangle}, \tag{3.8}$$

where ϕ' is the fluctuation of any of the primitive variables, $\mathbf{x} = (x, y, z)$, and $\langle\cdot\rangle$ denotes averaging in time. In Figure 3.12, the $\Re_{\phi\phi}$ results are displayed for P1 and P2.

The two-point correlation functions of the velocity components tend rapidly to zero at both probe positions when approaching the midspan, which indicates a sufficient spanwise length. On the other hand, $\Re_{pp}$ needs longer distances compared to the velocity functions to obtain uncorrelated values. Nonetheless, its value is reasonably low at $z = 0.15c$.

The mesh has been generated by extruding a 2D grid in 160 layers along the spanwise direction (z). The plane (x, y) mesh, is an all-triangle unstructured grid except for the wall adjacent elements which are square-shaped to control the first off-wall node distances properly. The 2D mesh has a resolution of 1.2×10^5 grid points; thus, the 3D mesh size is around 19.2×10^6 CVs. Since no experimental or WRLES data on boundary layer thickness was available, a preliminary LES computation was performed to estimate δ, being the smallest value approximately $\delta_{min} \approx 0.01c$ located at the airfoil leading edge. The quick transition to turbulence of this airfoil makes the boundary layer to thicken rapidly, reaching relatively high values of δ very close to the leading edge. The first off-wall LES nodes were placed at a wall distance of $y_1 = 0.0025c$, which is well inside the boundary layer but trying not to harm

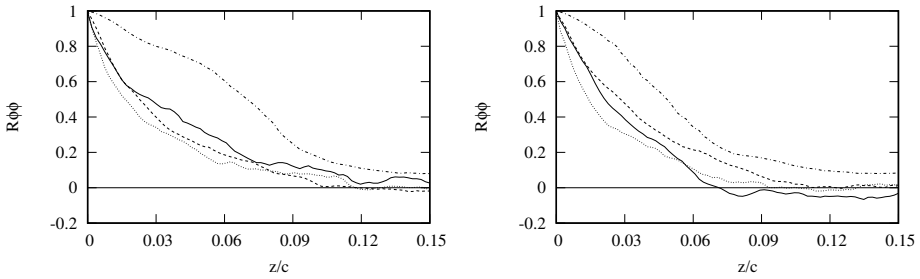

Figure 3.12: Two-point correlation of the three velocity components and pressure fluctuations along the spanwise direction. P1 left, and P2 right (see Figure 3.11 for probe positions). $\Re_{uu}$ (·····), $\Re_{vv}$ (————), $\Re_{ww}$ (- - - - -) and $\Re_{pp}$ (-·-·-·-·-).

excessively the time-step size, Δt. The LES grid resolution parameters used by Park and Moin to perform WMLES of a NACA4412 airfoil[9] with mild flow separation have been followed. In their computation, the authors used space-averaged grid spacings of $\overline{\Delta x}^+ \approx 160$ and $\overline{\Delta z}^+ \approx 62$. On the other hand, the WM mesh has been generated by extruding 30 layers from the solid surface up to the first off-wall nodes at $y_1 = 0.0025c$. The layers are conveniently concentrated towards the wall to get an appropriate first off-wall node distance (i.e., $\Delta y_1^+ < 1$).

In Figure3.13, the grid spacings in the wall-normal and the two wall-parallel directions are displayed for both, the WM and the LES mesh. It has to be taken into account that the wall-parallel grid spacings are common for both grids. The local shear stress provided by the WM has been used to normalize the grid distances as much accurately as possible. The LES mesh has a poor resolution since the minimum value of Δy_1^+ is approximately 18. This is far away from the viscous sublayer, being impossible to capture the real flow dynamics, especially in the leading edge region where Δy_1^+ reaches a value of almost 120. On the other hand, the WM mesh has a good resolution, with a Δy_1^+ below 0.4 throughout the solid surface. Regarding the grid spacing in the other directions, the stream and the spanwise resolutions range between $40 \lesssim \Delta x^+ \lesssim 200$ and $15 \lesssim \Delta z^+ \lesssim 80$, which is below the limits used by Park and Moin [9].

Finally, in Figure 3.14, the ratio of the grid size with respect to the Kolmogorov scales ($\Gamma = h/\eta$) is displayed in order to ensure that the LES model is working within its range of applicability in the far field region. According to Pope [39], in order to not lose accuracy in numerical predicitons, the spatial filter size h should be smaller than the characteristic sizes of flow structures in the intertial/dissipation range limit. This corresponds to a ratio of the grid spacing with the Kolmogorov

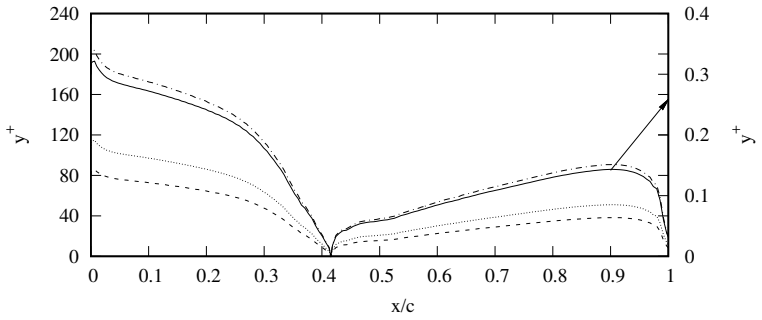

Figure 3.13: Left axis: Distributions of Δy_1^+ (·····), Δx^+ (-·-·-·) and Δz^+ (- - - - -) of the LES mesh along the upper surface of a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and $AoA = 15.2°$. Right axis: WM mesh Δy_1^+ (————).

scales of approximately $\Gamma < 40$ (See Chapter 2).

Regarding the boundary conditions, a uniform velocity profile of magnitude u_∞ is set with the appropriate angle of attack at the inflow ($u = u_\infty cos(AoA)$, $v = u_\infty sin(AoA)$ and $w = 0.0$) while at the outflow boundary, a pressure-based condition is imposed. No-slip conditions on the airfoil surface are prescribed and finally, in the spanwise direction, periodic boundary conditions are used.

The same procedure as in previous sections has been followed to determine the temporal filtering periods to carry out the present study. The power spectrum of the streamwise velocity component has been obtained in the near-wall area within the boundary layer detached region. Its graph is displayed in Figure 3.15 on the right, together with the selected TAF cut-off frequencies. In Table 3.5, the different TAF configurations are summarized. Time is measured in non-dimensional time units defined as $c/u_\infty = 1\,TU$. Unlike previous tests, the flow shows a significant large-scale unsteady behavior due to the boundary layer detachment/reattachment process, whose characteristic time-scale will be denoted as T_{dr}. Thus, the largest filtering period is selected to assess the effects of suppressing the contribution of the biggest flow motions to the wall shear stress in non-equilibrium unsteady conditions.

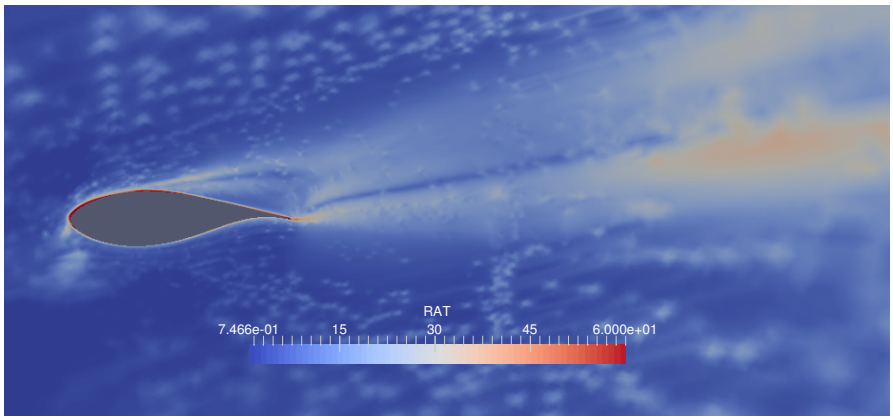

Figure 3.14: Ratio between the grid size and the Kolmogorov scales ($\Gamma = h/\eta$) of a flow past a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and $AoA = 15.2°$.

Table 3.5: Summary of the different TAF configurations tested with the flow around a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and $AoA = 15.2°$.

Config.(n)	Filter length T_n	$f_n = 1/T_n$	Energy spectrum range
0	no filter	no filter	N/A
1	0.1	10.0	inertial/dissipation range limit
2	4.0	0.25	inertial/energy-containing range limit ($T_2 \approx 0.27T_{dr}$)
3	10.0	0.1	large fraction of the largest time-scale ($T_3 \approx 0.7T_{dr}$)

The time-averaged statistics have been collected along $80\,TU$ with a transient period of $30\,TU$. These intervals were obtained by analyzing the instantaneous lift coefficient shown in Figure 3.15 on the left, corresponding to the simulation performed with a filtering period of $T_2 = 4.0$. According to the $C_l(t)$ chart, the initial transient perturbations vanish at $t \approx 30\,TU$. On the other hand, since the instantaneous C_l is closely related to the detachment/reattachment process of the boundary layer, the power spectrum of its signal has been used to determine the characteristic time-scale of the largest and most energetic motions, T_{dr}, whose value is $14.49\,TU$. Therefore, the total averaging period of $80\,TU$ represents approximately six detachment/reattachment cycles.

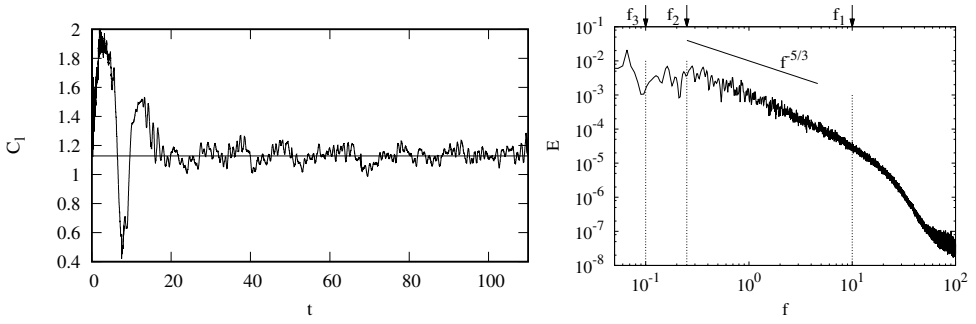

Figure 3.15: Left: Instantaneus lift vs. time in dimensionless TU. The horizontal continuous line represents the experimental average value[33] of $C_l = 1.128$. Right: Energy spectrum of the streamwise velocity component obtained at a near-wall position in the recirculation region.

Based on the preliminary LES computation used for the approximation of δ, an estimation of the grid resolution needed at the wall for a WRLES was done. This allowed making a projection of the diffusive CFL upper bound for the time-step as well as of the total number of grid points, N_{cv}, resulting in $dt \approx 4 \times 10^{-6}$ and $N_{cv} \approx 100$ million, respectively. This is one order of magnitude more regarding the grid size and two orders of magnitude for the time-step, compared to the WMLES. These values are in line with other studies [9].

Test results

In Figure 3.16, LES-only and WMLES results of the time-averaged pressure coefficient, $C_p = 2(p - p_\infty)/\rho u_\infty^2$, are shown together with experimental data of Timmer and van Rooij [33]. WMLES computations have been carried out according to the TAF configurations summarized in Table 3.5. For the sake of clarity, configuration 1 results are not shown since they are very similar to those of configuration 3.

On the other hand, in Figure 3.17, numerical results of skin friction distribution in the streamwise direction, $C_{fx} = 2\tau_{wx}/\rho u_\infty^2$, obtained in the same conditions as the C_p, are plotted for the upper surface.

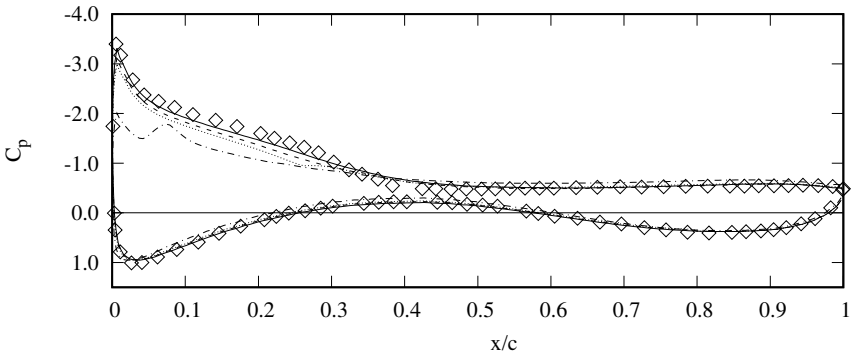

Figure 3.16: Pressure coefficient distribution along the upper and lower surface of a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and at $AoA = 15.2°$. LES-only results (-·-·-·), non-filtered WMLES (·····), WMLES with $T_2 = 4.0$ (————), and WMLES with $T_3 = 10.0$ (- - - - -). Experimental data [33] (◇).

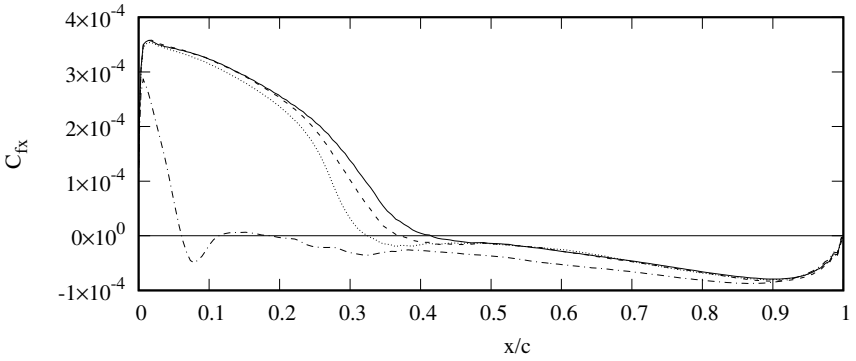

Figure 3.17: Skin friction coefficient distribution in the streamwise direction, C_{fx}, along the upper surface of a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and at $AoA = 15.2°$. LES-only results (-·-·-·), non-filtered WMLES (·····), WMLES with $T_2 = 4.0$ (————), and WMLES with $T_3 = 10.0$ (- - - - -).

Good agreement between experimental and numerical results are obtained for the C_p when the WM filtering period is set at $T_2 = 4.0$. This filter length corresponds to the energy-containing/inertial range limit while representing a relatively small

fraction of the largest flow time-scale, T_{dr}, specifically, $T_2 \approx 0.27T_{dr}$. As in the Pipe Flow tests, poor results are obtained if higher frequencies are not filtered. However, unlike in equilibrium conditions, deficient predictions are also obtained when the lowest frequencies associated with the largest and most energetic flow scales are suppressed. Specifically, for $T_3 = 10.0$, which is approximately 0.7 times T_{dr}, the numerical C_p becomes underpredicted with respect to the experimental reference.

Regarding the skin friction coefficient, no experimental data is available for comparison. Nonetheless, the C_{fx} curve obtained with a filter length of $T_2 = 4.0$ shows a consistent behavior with respect to the experimental C_p. In this case, the skin friction becomes negative at the same position where the experimental C_p curve shows the beginning of the pressure plateau (i.e., around $x/c \approx 0.43$). These two particularities in the C_{fx} and C_p curves are associated with the boundary layer detachment point, which suggests that the C_{fx} is well predicted at least from a qualitative point of view.

The consequences of using an inadequate filtering period seem to be concentrated in the flow detachment point region. Significant discrepancies in the C_{fx} predictions are observed in this region when different averaging periods are used, while almost identical results are obtained towards the airfoil edges. This is because the instantaneous position of the detachment point varies significantly in time due to the boundary layer detachment/reattachment process. This causes a strongly unsteady behavior of the instantaneous skin friction in that region, with a characteristic fluctuating period of T_{dr}. By contrast, at the airfoil ends, the skin friction seems to have a more constant value, at least in the leading edge area, making the results in these regions more insensitive to T. These observations suggest that the unsteady component of the skin friction derived from very-large-scale motions play an essential role on the flow global behavior. A sufficiently small averaging period allowing to capture these effects must be used.

On the other hand, in Table 3.6, the integral values of the drag and the lift coefficients ($C_d = 2D/\rho u_\infty A$, $C_l = 2L/\rho u_\infty A$, where D and L are the total aerodynamic forces in the stream-normal and the streamwise directions and A is the airfoil surface) are presented and compared with the experimental data. For the drag coefficient, the viscous and the pressure components are shown together with its total value. The numerical results are obtained in the same conditions as the C_p and C_{fx} plots presented above. The conclusions derived from the pressure and skin friction coefficients are confirmed with the integral magnitudes. A slight discrepancy between the numerical C_d at $T_2 = 4.0$ and the experimental reference is observed. According to the two-point correlations in Figure 3.12, there is no need for a wider computational domain. Thus, the remaining discrepancy could be caused by the geometrical differences between the experimental and the numerical domains which feature completely different external boundaries.

Table 3.6: Lift and drag coefficient values of a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and at $AoA = 15.2^\circ$ obtained with and without WM together with experimental data [33]. The WMLES results have been obtained with the TAF configurations summarized in Table 3.5. The relative differences between the numerical and the experimental data are shown in %.

Model	Filter-length T	C_l	rel. diff. [%]	C_d (press.)	C_d (visc.)	C_d (total)	rel. diff. [%]
Experimental	N/A	1.128	–	–	–	0.1144	–
LES-only	N/A	0.932	17.37	0.2063	1.79×10^{-4}	0.2064	80.41
WMLES	No Filter	1.030	8.68	0.1503	2.30×10^{-4}	0.1505	31.55
WMLES	0.1	1.080	4.25	0.1353	2.46×10^{-4}	0.1355	18.44
WMLES	4.0	1.123	0.44	0.1298	2.52×10^{-4}	0.1300	13.63
WMLES	10.0	1.061	6.02	0.1408	2.44×10^{-4}	0.1410	23.25

In Table 3.7, the boundary layer detachment/reattachment characteristic periods, T_{dr}, are shown. The objective is to analyze the influence of the wall shear stress temporal resolution on the unsteady dynamics of large-scale motions. According to obtained results, the characteristic periods are affected by the filter length T. Nonetheless, it is unclear whether the dispersion of T_{dr} results is a consequence of a poorly resolved mean flow, or the averaging period directly conditions its value. Although there is a correlation between T and T_{dr} (the larger the filter length, the larger the flow characteristic time-scale), by comparing LES-only and WMLES results, it can be concluded that having a well-predicted mean flow strongly influences the T_{dr} value.

Table 3.7: Detachment/reattachment characteristic time-scales (T_{dr}) of a flow around a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and $AoA = 15.2^\circ$. Results are obtained for LES-only and WMLES computations. In case of WMLES the TAF configurations in Table 3.5 have been used.

Model	Filter-length T	Char. time-scale T_{dr}
LES-only	N/A	8.33
WMLES	No Filter	12.25
WMLES	0.1	12.34
WMLES	4.0	14.49
WMLES	10.0	15.87

Regarding the mean flow behavior, in Figure 3.18, streamlines are depicted for the LES-only and the WMLES with $T_2 = 4.0$ configurations. According to the skin friction plots, the boundary layer detachment point is located at $x/c \approx 0.22$ in the LES-only and at $x/c \approx 0.43$ in the WMLES test. The skin friction magnitude at the leading edge and the detachment point location are related. A rising value of C_f in

this region, entails a backward displacement of the separation point. This effect is well reflected in the plots, in which the trailing edge recirculation bubble shortens as the skin friction rises.

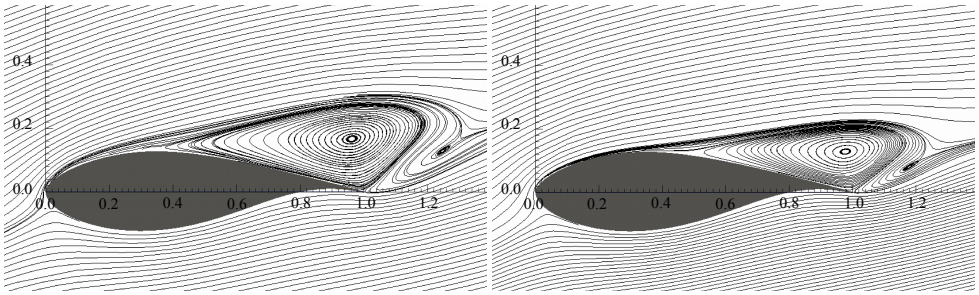

Figure 3.18: Mean flow streamlines around DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and at $AoA = 15.2^\circ$ obtained without (left) and with WM with $T_2 = 4.0$ (right).

Finally, in Figure 3.19, isosurfaces of the non-dimensional second invariant of the velocity gradient tensor, Q, are displayed at $Q = 1$ for the WMLES case. Streamwise turbulent structures emerge from the leading edge of the airfoil, and thus, no laminar behavior is observed in this region. This confirms that the wall model is working on a turbulent boundary layer all along the airfoil surface, and therefore, it is in its range of applicability.

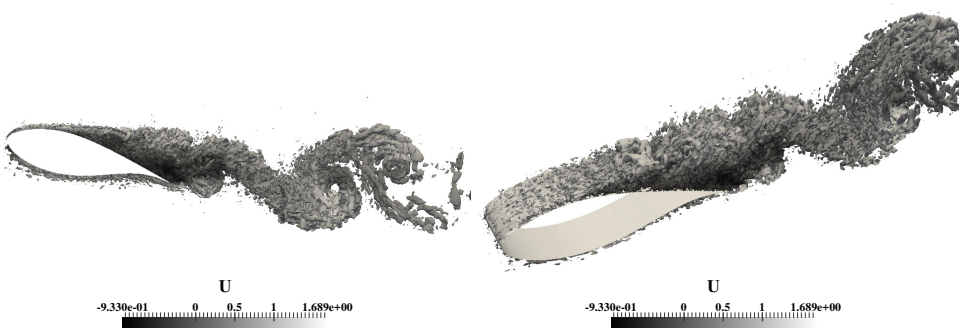

Figure 3.19: Instantaneous flow structures of a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ and at $AoA = 15.2^\circ$ obtained with WM with $T_2 = 4.0$. The structures are visualized through isosurfaces of the second invariant of the velocity gradient tensor with a non-dimensional value of $Q = 1$ and colored by the streamwise velocity magnitude.

3.4.3 Ahmed Car test at $Re = 7.68 \times 10^5$. Application of TLM to fully 3D geometries.

Test description

In this section, the flow around an Ahmed car geometry at a moderate Reynolds number of $Re = U_{ref}h/\nu = 7.68 \times 10^5$ based on the free stream velocity, U_{ref}, and the car height, h, is considered. This case aims at testing the WM in a fully 3D geometry which features sharp geometrical discontinuities. Although non-equilibrium conditions such as adverse pressure gradients are also present, in this case, the boundary layer detachment at the back car surface is mainly caused by its geometry.

In Figure 3.21, details of the computational domain and the car geometry are shown. The numerical tests are performed in a rectangular box of $9.1944 \times 1.87 \times 1.4m$ where the x, y, z are the streamwise, the side boundary-normal and the bottom-normal directions, respectively. The car front is placed at $2.2014m$ downwind of the inlet boundary. Regarding the car dimensions, the length is $1.044m$, the height is $0.288m$ while the back surface slant angle is 25°.

The boundary conditions are shown in Figure 3.21. Namely, a uniform velocity of magnitude U_{ref} is prescribed at the inlet boundary, $\mathbf{u} = (U_{ref}, 0, 0)$, while a pressure based condition is set at the outlet surface. As the experimental setup is a $3/4$ open wind tunnel, slip wall conditions are imposed at the side and top boundaries. Finally, no-slip conditions are applied to the car surface and the bottom boundary.

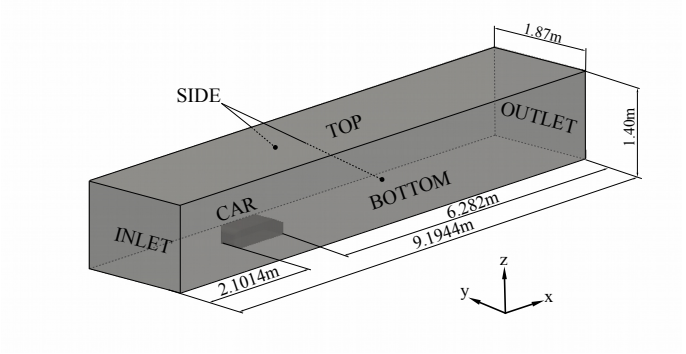

Figure 3.20: Computational domain of the Ahmed car body test.

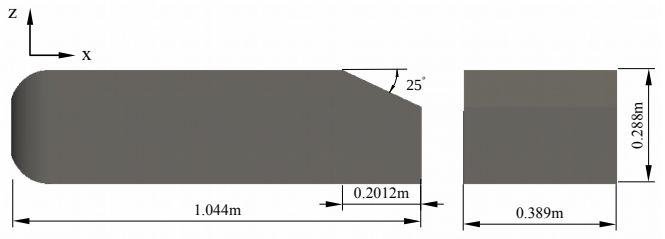

Figure 3.21: Ahmed car body geometry.

The experimental data of Lienhart *et al.*[40] is used for comparison, and thus, the experimental arrangement is reproduced numerically. The experimental results concern the time-averaged streamwise velocity and rms velocity fluctuation profiles on the top and the slanted surfaces as well as the integral pressure drag coefficient.

In a previous work of Aljure *et al.*[21], LES-only tests were performed with a hybrid tetra-prism mesh with five prismatic layers adjacent to the solid wall and a grid resolution of $1.56 \times 10^6 CV$ with a $\Delta y_1 = 0.0005m$. The results obtained in these computations were not far from the experimental reference although they could be slightly improved in some areas, especially in the car top surface region. The mesh parameters of Aljure *et al.* have been used to design a more challenging mesh (i.e., a coarser mesh with higher Δy_1) for the WMLES to have a wider improvement margin respect to the LES-only results. This new mesh is an all-tetrahedron unstructured grid with 4.8×10^5 control volumes and a $\Delta y_1 = 0.002m$. Only the first row of

control volumes adjacent to the solid surface are prisms which have been generated by extruding the all-triangle unstructured superficial mesh. This strategy has been used to control the first off-wall node distance properly. The new mesh resolution is three times coarser than Aljure's one, while the first off-wall node separation is four times larger than in the finer grid. In both cases, the control volumes are mainly concentrated near the car surfaces especially in the recirculation region. In order to estimate the boundary layer thickness, LES results from Aljure *et al.*[21] has been used. The smallest value of δ throughout the upper surface of the car is approximately $\delta_{min} \approx 0.02m$. Nonetheless, a very thin flow recirculation in the back surface of the car sets an even more restrictive criterion to determine the first off-wall node height. By inspecting the mean flow streamlines of Aljure's test (see Figure 3.27a in $-0.16m \lesssim x \lesssim -0.07m$), the flow recirculation height is determined at $0.006m$ which is much thinner than δ on the upper surface. This thin recirculation, limits the coarseness of the rough challenging mesh. At least, two points must be placed in the LES mesh to fulfill the Nyquist criterion, and thus, to capture the flow characteristic vortex[8]. Otherwise, it is not possible to predict the near-wall flow field properly even using the WM, according to the mesh convergence study carried out initially.

The WM will be applied to the coarse and to the fine mesh of Aljure *et al.*. The WM meshes will be extruded at the distances of $h_{wm} = 0.002m$ and $h_{wm} = 0.0005m$ respectively, matching the first off-wall row of nodes in both cases. The number of extrusion layers will be 15 for the coarse and 8 for the finer mesh, and they will be concentrated towards the wall following a hyperbolic tangent distribution law.

In Figure 3.22, the first off-wall node distances (Δy_1^+) as well as the wall-parallel characteristic length (Δx_w^+) in wall units, are compared for the fine and the coarse LES meshes. The wall-parallel characteristic length is defined as $\Delta x_w = \sqrt{S_w}$, where S_w is the area of the wall-adjacent face of a LES control volumes. For the sake of clarity, the WM mesh Δy_1^+ is plotted only for the fine mesh case since it is very similar to the coarse mesh one. The wall shear stress obtained with the WM has been used to normalize the distances plotted in the figure.

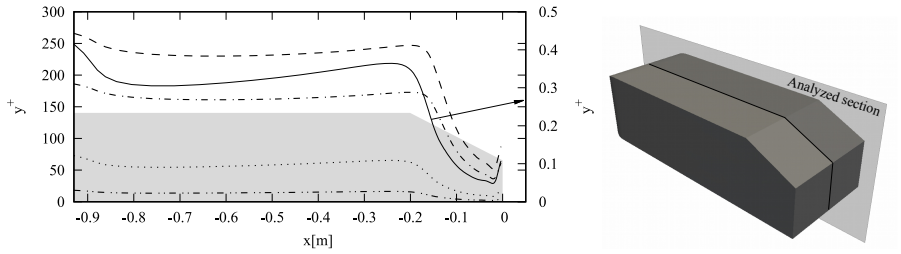

Figure 3.22: Left axis: Distributions of Δx_w^+ and Δy_1^+ along the upper surface of the Ahmed car geometry at $Re = 7.68 \times 10^5$. Coarse mesh: Δx_w^+ (- - - - - -) and Δy_1^+ (.....). Fine mesh: Δx_w^+ (- · - · - ·) and Δy_1^+ (- ·· - ·· - ··). Right axis: WM mesh Δy_1^+ (————).

The main parameters of the two grids are summarized in the table below:

Table 3.8: Mesh parameters of the coarse mesh and the finer grid of Aljure *et al.*[21].

Mesh name	Grid resolution	Δy_1	Δy_1^+ range	Δx_w	Δx_w^+ range
Aljure's fine mesh	1.56×10^6 CV	$0.0005m$	$[1-18]$	$\approx 0.007m$	$[35-185]$
Coarse mesh	4.8×10^5 CV	$0.002m$	$[5-72]$	$\approx 0.01m$	$[50-256]$

The computational strategy proposed by Aljure *et al.*[21] has been followed to analyze the WM effects under similar conditions. Thus, the WALE model has been used as a subgrid strategy while the time-averaged statistics were gathered over 70 time units ($tU_{ref}/h = 1\,TU$) after a transient period of $10\,TU$.
On the other hand, to determine the time averaging period, the power spectrum of the streamwise velocity component has been obtained in the near-wall area in the recirculation region. The energy containing/intertial range limit is at $T = 3\,TU$ which is approximately the time that takes for a particle to go all the way through the car upper surface. Finally, Q isosurfaces have been obtained form Aljure's LES data [21] revealing that the flow is turbulent throughout the car surface except in the front area, where the WM was explicitly switched off.

Test results

In Figure 3.23, the normalized mean streamwise velocity (U/U_{ref}) at the car symmetry plane is plotted for the experimental results of Lienhart *et al.*[40], the LES results of

Aljure *et al.*[21] and the WMLES data obtained with the same mesh. The results obtained with the fine mesh of Aljure *et al.*[21], show fairly good agreement with the experimental data both, in the LES-only and the WMLES configurations. However, the WM produced slight improvements, especially on the car's top surface which induced the improvements downstream. In Figure 3.24, the normalized streamwise velocity is plotted again for the coarse mesh. A more significant improvement of the LES-only results is observed when applying the WM. Nonetheless, given the grid coarseness, the final results are not as satisfactory as the WMLES results obtained with Aljure's mesh. Some discrepancies are observed between the fine and the coarse mesh WMLES results in the thin recirculation region (see Figures 3.28b and 3.27b between $-0.16m \lesssim x \lesssim -0.07m$). This is due to the poor resolution of the coarse LES mesh in this area, in which only two LES points were placed in the wall-normal direction. The use of the WM, despite improving the results in most regions, it is not able to overcome the lack of minimum LES resolution in this specific area. However, a significant improvement is achieved again in the car top surface which is of great importance for the correct development of the flow downstream.

Regarding the time-step, there are no significant differences between the coarse and the fine LES meshes. Due to the relatively low Reynolds number, and the relatively similar wall-parallel grid spacing (see Figure 3.22), the time-step size of both cases is almost identical being its value around $2 \times 10^{-4}\, TU$. This shows that the use of shear stress wall models at low and moderate Reynolds number, is much less profitable than at high Reynolds number since the gains derived from the time-step size increase are less significant.

In Figures 3.25 and 3.26, the normalized streamwise velocity fluctuations (U_{rms}/U_{ref}) are shown for the same tests and configurations than in the previous plots. In this case, the effects of the WM seem to be more pronounced, especially in the near-wall region. It seems that unlike the pipe flow test, the LES mesh resolution in that area is fine enough to capture some near-wall flow structures resulting in fairly good predictions which are enhanced by the WM.

Streamlines of the mean flow on the car symmetry plane are depicted in Figure 3.27 for the fine grid, and in Figure 3.28 for the coarse mesh, respectively. In both plots, the LES-only (a) and WMLES (b) configurations are displayed together. In the coarse mesh tests, the plot corresponding to the LES-only computation (Figure 3.28a), it reveals a secondary recirculation in the rear part of the slanted surface which is not observed in Aljure's test (Figure 3.27a). In the latter case, the mean velocity profiles match the experimental reference significantly better than in the coarse mesh test. The use of the WM suppresses this recirculation probably because of the better predicted upstream inflow. These observations are coherent with the mean streamwise velocity profiles shown in Figure 3.24. A general improvement of the results is observed when using the WM, but especially strong corrections are

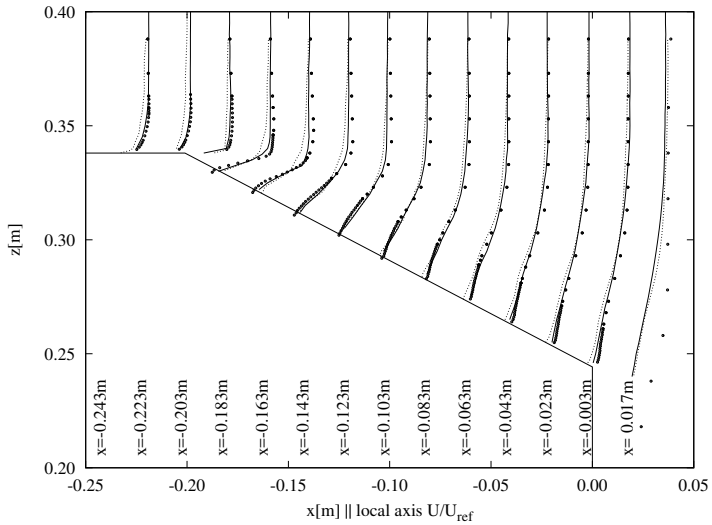

Figure 3.23: Normalized mean streamwise velocity component at the car symmetry plane on the Ahmed car back surface at $Re = 7.68 \times 10^5$. LES-only (·····) and WMLES (————) data, both obtained with Aljure's fine mesh [21]. Experimental data of Lienhart *et al.*[40] (•).

found on the car top surface ($x \lesssim -0.2m$) and in the lower part of the slanted surface ($x \gtrsim -0.07m$) due to the elimination of the secondary recirculation bubble. On the other hand, in the fine mesh case results displayed in Figure 3.27, no significant deviations are revealed between the LES-only and the WMLES plots which is in line with the slight differences observed in the mean flow velocity profiles in Figures 3.23a and 3.23b.

Finally, in Table 3.9, the pressure and viscous drag (C_{dp} and C_{dv}) and lift (C_{lp} and C_{lv}) coefficients are shown for both, the coarse and the fine mesh, with and without WM. Only experimental data on the pressure drag is available. In parenthesis, the relative differences between the experimental results and all the numerical data are shown. Good agreement between the experimental and the WMLES results of the C_{dp} is obtained. Although the C_{dp} is an integral magnitude, the improvement of the WMLES with respect to the LES-only configuration is mainly due to the better resolution of the pressure field in the rear part of the car. In the front part, which has an important weight in the total C_{dp} value, the WM was switched off due to the laminar behavior of the flow. This allows concluding that the WM caused an improvement

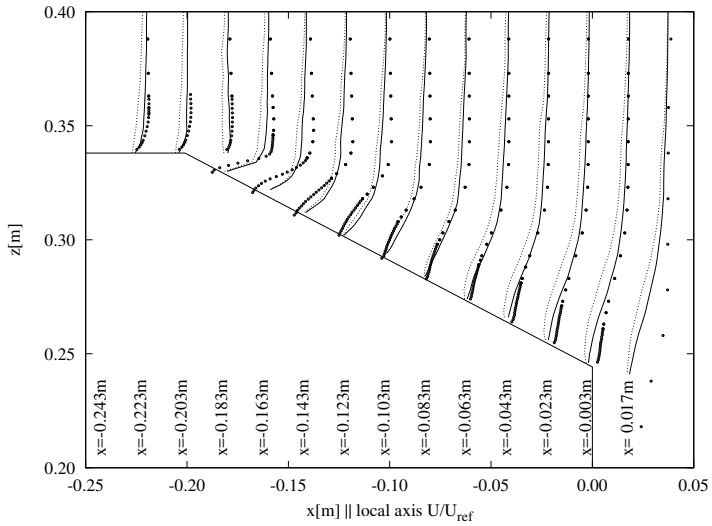

Figure 3.24: Normalized mean streamwise velocity component at the car symmetry plane on the Ahmed car back surface at $Re = 7.68 \times 10^5$. LES-only (·····) and WMLES (————) data, both obtained with the coarse mesh of 4.8×10^5 grid points. Experimental data of Lienhart *et al.*[40] (•).

in the resolution of the relevant part of the flow, where complex non-equilibrium phenomena are present.

Regarding the other magnitudes, the integral viscous forces represented by C_{dv} and C_{lv} obtained with the WM are higher than their LES-only counterparts which indicates that the wall shear stress is underpredicted when not using the WM.

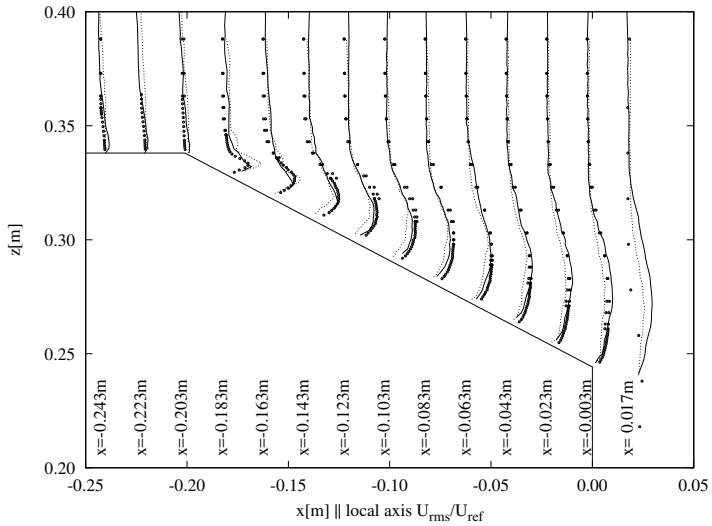

Figure 3.25: Normalized mean streamwise velocity fluctuations at the car symmetry plane on the Ahmed car back surface at $Re = 7.68 \times 10^5$. LES-only (·····) and WMLES (————) data, both obtained with Aljure's fine mesh [21]. Experimental data of Lienhart *et al.*[40] (•).

Table 3.9: C_{dp}, C_{dv}, C_{lp} and C_{lv} of the Ahmed car at $Re = 7.68 \times 10^5$. Experimental results by Lienhart *et al.*[40] regarding C_{dp} are displayed together with numerical results obtained with two different meshes (see Table 3.8 for mesh details) both, with and without WM. In parenthesis, the relative differences of the numerical data with respect to the experimental results are given in %.

	C_{dp}		C_{dv} $(\times 10^{-3})$	C_{lp}	$C_{lv}(\times 10^{-5})$
Lienhart *et al.*[40]	0.285	(−)	−	−	−
coarse LES-only	0.322	(12.98)	0.305	0.0927	−4.92
coarse WMLES	0.297	(4.21)	3.12	0.1681	4.562
Aljure *et al.*[21] LES-only	0.304	(6.67)	1.23	0.257	5.78
Aljure's mesh WMLES	0.294	(3.15)	2.31	0.241	7.49

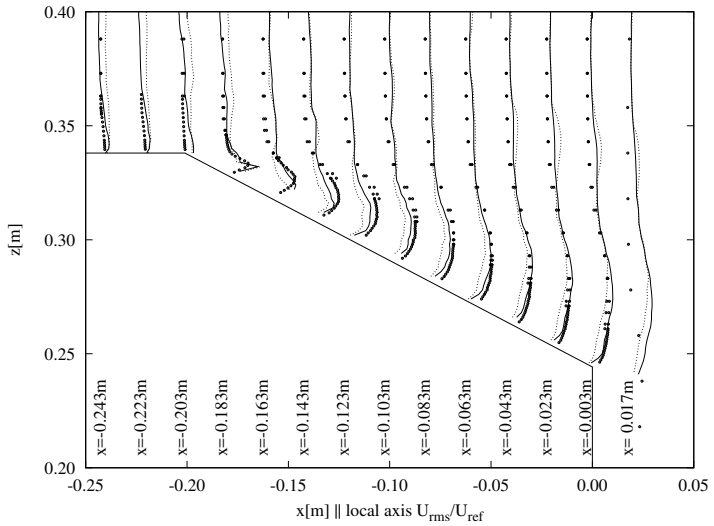

Figure 3.26: Normalized mean streamwise velocity fluctuations at the car symmetry plane on the Ahmed car back surface at $Re = 7.68 \times 10^5$. LES-only (·····) and WMLES (————) data, both obtained with the coarse mesh of 4.8×10^5 grid points. Experimental data of Lienhart *et al.*[40] (•).

3.5 Analysis of the interaction between the LES and the WM.

In this section, the interdependence between the present TLM formulation and the subgrid LES model is evaluated. In tests presented above, as well as in most of the previous works in the literature[7, 8, 9, 13], the dynamic Smagorinsky formulation has been used as a subgrid model, and to the author best knowledge, the interaction of a TLM with other LES models has not been assessed yet. The analysis will be carried out by computing the Pipe flow at $Re_\tau \approx 3000$ test with five different LES models with different dissipative properties and near-wall behaviors.

Firstly, the standard Smagorinsky[41] (SMG) model with no modifications will be used. In this model, the subgrid viscosity is considered directly proportional to the local strain, and its behavior near the wall is completely unphysical since ν_{sgs} does not vanish towards the wall. The Germano dynamic modification [18] of the Smagorinsky model (DSMG) addresses this deficiency without requiring damping

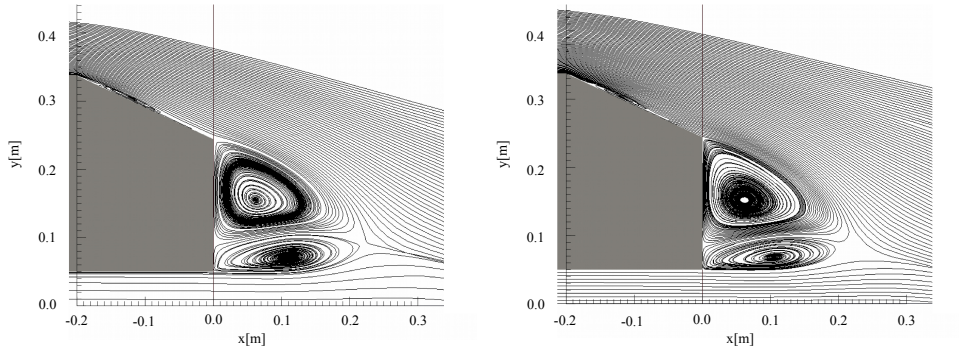

Figure 3.27: Mean flow streamlines of the Ahmed car at $Re = 7.68 \times 10^5$ on the car symmetry plane obtained with the Aljure's fine LES mesh [21]. (a) LES-only test (b) WMLES test.

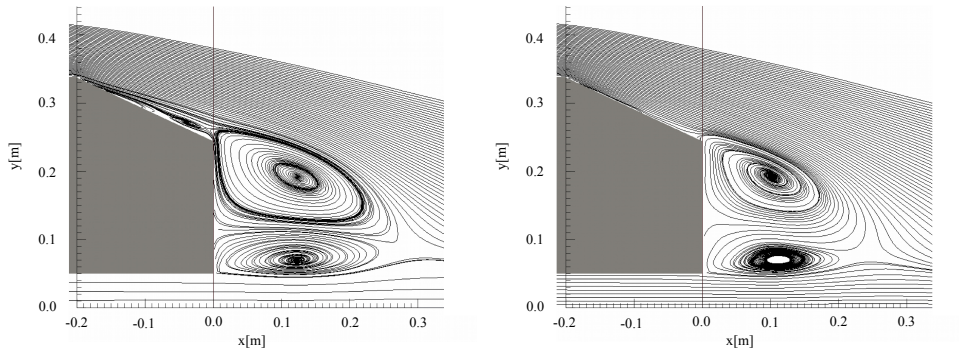

Figure 3.28: Mean flow streamlines of the Ahmed car at $Re = 7.68 \times 10^5$ on the car symmetry plane obtained with the coarse LES mesh. (a) LES-only test (b) WMLES test.

functions. Germano showed that the new model has an asymptotic behavior of $\mathcal{O}(y^3)$ for the SGS stresses in the wall area. More recent developments led to new LES models which intend to improve the physical basis of the existing formulations. This is the case of the wall-adaptive local eddy viscosity (WALE) formulated by

Nicoud *et al.*[22] in which the subgrid viscosity is a function not only of the strain but also of the rotation rate, as Wray and Hunt [42] observed in different DNS of isotropic turbulence. Moreover, the model also shows an $\mathcal{O}(y^3)$ asymptotic behavior of the subgrid viscosity in the near-wall region while ν_{sgs} almost vanishes in pure-shear regions, providing accurate laminar to turbulent transitions since it allows instabilities in the laminar region. The variational multiscale (VMS) approach for LES is also of interest since only the effects of the small-scale velocity field are taken into account when modeling the subgrid stress tensor. This improves the model damping outcome given that the large scales are well resolved, and the subgrid viscosity only takes into account the under-resolved small-scale effects. An a priori scale separation is performed by explicitly filtering the resolved velocity field, and then, only the small-scale resolved part of the flow is used to evaluate ν_{sgs} by means of a standard LES model. The original formulation was proposed by Hugues [43], and the Smagorinsky model was used to close the formulation. In the present work, the ν_{sgs} is obtained through the WALE model (VMS-WALE) [44, 45] to take advantage of the properties commented above. Finally, a newer approach named QR model proposed by Verstappen [46] will also be tested. This model is derived from the premise that only the strictly necessary level of dissipation must be provided. A lower bound for ν_{sgs} is found by requiring a sufficient damping level that prevents the non-linear mechanisms of the convective term from producing eddies of any size smaller than the grid size.

3.5.1 Test description

The test setup is exactly the same as the Pipe flow at $Re_\tau \approx 3000$ test reported at Section 3.4.1. Firstly a LES-only computation is performed to assess the improvements introduced by the wall model. Then, four different wall model configurations are tested, a not-filtered setup with extrusion heights of $h^+_{wm} \approx 30$ and $h^+_{wm} \approx 210$, and a filtered configuration with filter length of $T = 10.0$ and the same extrusion heights. The selected filter length of $T = 10.0$ corresponds to the filter width that minimizes the computed wall shear stress error according to the initial Pipe flow experiment. On the other hand, the five LES models described above will be used to account for the unresolved subgrid scales effects.

3.5.2 Test results

In Table 3.10, the computed Re_τ is compared for the five different LES models. For each subgrid model, five different wall strategies have been applied, LES-only, and filtered and non-filtered WM at $h^+_{wm} \approx 30$ and $h^+_{wm} \approx 210$. In case of filtered configuration, the filter length is set at $T = 10.0$. In brackets, the relative error of the computed Reynolds number with respect to the reference value of $Re_\tau \approx 3026$ is shown.

Table 3.10: Computed Re_τ values of the pipe flow at $Re = 1.33 \times 10^5$ which corresponds to $Re_\tau \approx 3026$. The values are obtained with and without WM for five different subgrid models. In case of WMLES, two different mesh heights are tested ($h^+_{wm} \approx 30$ and $h^+_{wm} \approx 210$) with and without TAF. The relative error in (%) with respect the reference value ($Re_\tau \approx 3026$) is in brackets.

	T	SMG ∇	DSMG $+$	WALE $\circ$	VMS-WALE $\triangle$	QR $\times$
No WM	N/A	954.1 (68.4)	1923.6 (36.4)	2189.3 (27.6)	1860.5 (38.51)	1343.4 (55.56)
WM $h^+_{wm} \approx 30$	N/A	3381.4 (11.74)	3409.2 (12.66)	3494.5 (15.48)	3541.8 (17.04)	3324.0 (9.84)
WM $h^+_{wm} \approx 210$		3281.4 (8.44)	3316.1 (9.58)	3423.0 (13.11)	3434.0 (13.48)	3241.4 (7.65)
WM $h^+_{wm} \approx 30$	10.0	3155.2 (4.26)	3135.6 (3.62)	3213.7 (6.20)	3172.1 (4.82)	2955.2 (2.33)
WM $h^+_{wm} \approx 210$		3096.1 (2.31)	3100.4 (2.45)	3167.7 (4.68)	3097.2 (2.35)	3070.2 (1.46)

Moreover, the values of Table 3.4 are also plotted in Figure 3.29 in a non-dimensional form ($Re_\tau / Re_{\tau_{ref}}$ being $Re_{\tau_{ref}} \approx 3026$). The symbols used in this figure are defined in Table 3.4.

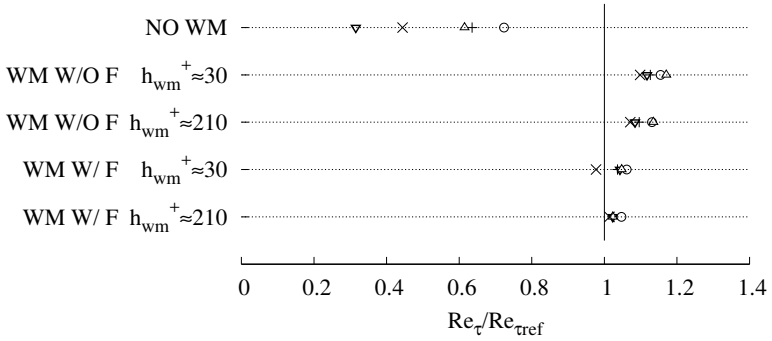

Figure 3.29: Non-dimensionl computed Re_τ values obtained with five different LES models, DSMG ($+$), QR ($\times$), SMG (∇), WALE ($\circ$), VMS-WALE ($\triangle$) and five different wall strategies: LES-only and WMLES at $h^+_{wm} \approx 30$ and $h^+_{wm} \approx 210$ with and without TAF. The reference value is $Re_{\tau_{ref}} \approx 3026$.

A great dispersion is observed in Figure 3.29 regarding the wall shear stress results obtained with the five different LES models when no WM is used. This was expected since the near-wall treatment of the proposed LES models differs significantly. Nonetheless, it appears that the WM is able to strongly reduce this

dispersion regardless the WM+LES combination used, showing that the particular near-wall behavior of each subgrid model is mostly overridden by the WM influence.

Regarding the mean streamwise velocity profiles, results of test cases shown in Table 3.10 are depicted in wall units and logarithmic scale in Figure 3.30. The data is distributed in five different charts, one for each LES model. All plots display results obtained with the five different wall treatments together with the DNS solution of Ahn *et al.*[32].

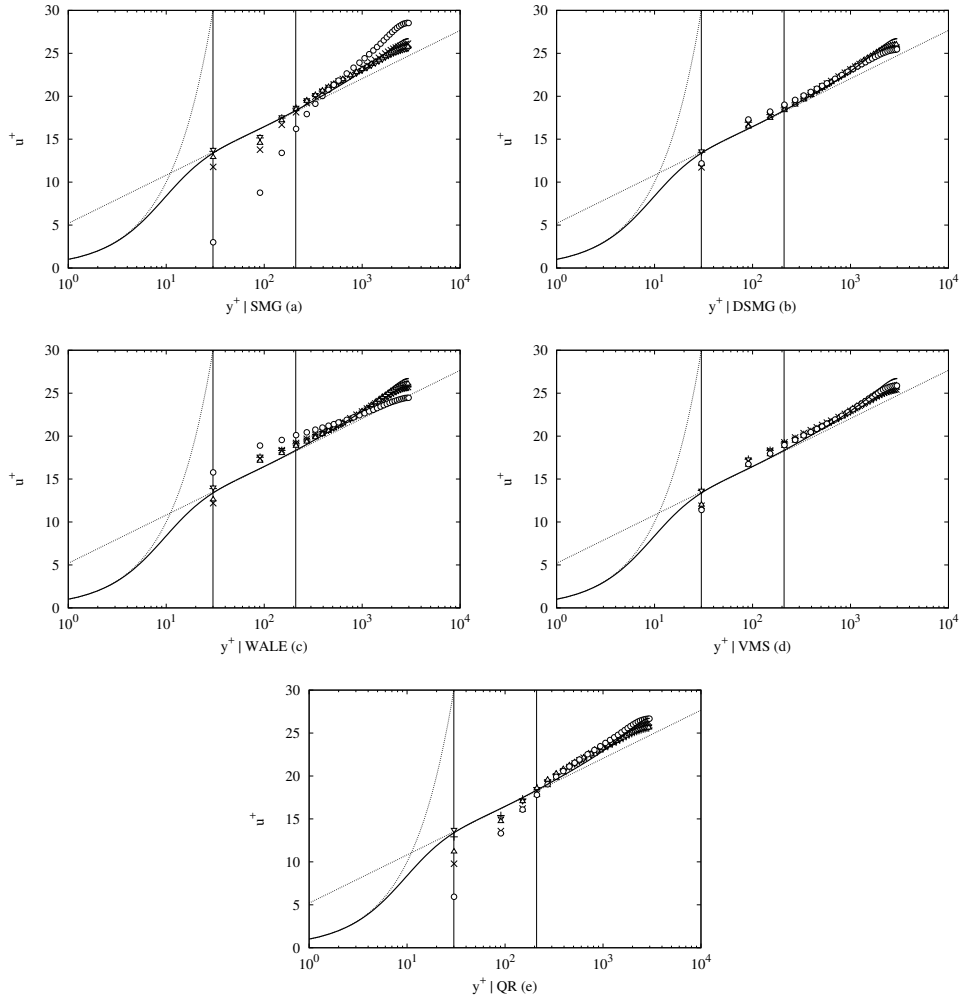

Figure 3.30: Mean streamwise velocity profiles in wall units . The wall distance is in logarithmic scale. The vertical lines represent the two LES/WM interface heights. The results have been obtained with five different LES models (a) SMG, (b) DSMG, (c) WALE, (d) VMS-WALE and (e) QR with five different wall treatments: LES-only (○), WMLES at $h_{wm}^{+} \approx 30$ with TAF (+) and without TAF(×) and WMLES at $h_{wm}^{+} \approx 210$ with TAF(▽) and without TAF(△). Lines conrrespond to DNS Ahn *et al.*[32] (————), and the law of the wall (·····).

In order to analyze in deeper detail the interaction of the LES and the WM, the mean streamwise velocity profiles are plotted again in Figure 3.31 but gathering the non-wall modeled LES tests in the figure on the left and the WMLES cases on the right. Additionally, in Figure 3.32, the same comparison is made for the rms of the velocity fluctuations in the streamwise, wall-normal and spanwise components as well as for the Reynolds shear stress. Taking into account the conclusions given above and for the sake of clarity, in the remaining plots of this section (Figures 3.31 and 3.32), only the time-filtered wall modeled tests with extrusion height of $h_{wm}^{+} \approx 30$ will be displayed since the differences with the $h_{wm}^{+} \approx 210$ results are negligible.

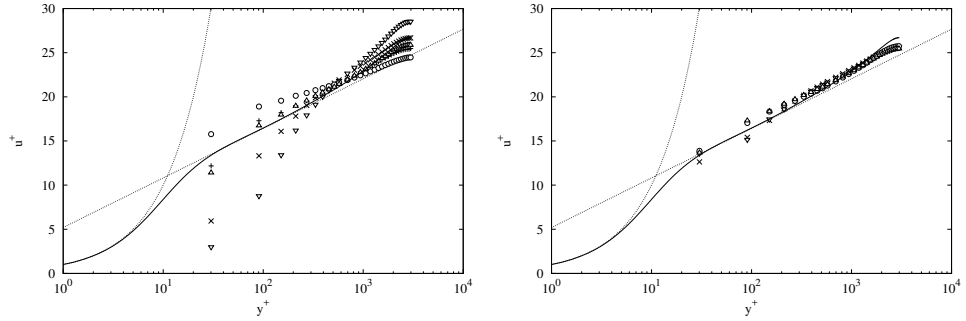

Figure 3.31: Mean streamwise velocity profiles in wall units . The wall distance is in logarithmic scale. The results have been obtained with five different LES models, DSMG (+), QR (×), SMG (▽), WALE (○), VMS-WALE (△) and two different wall strategies: LES-only (left) and WMLES at $h_{wm}^{+} \approx 30$ (right). DNS Ahn *et al.*[32] (————) and law of the wall (·····).

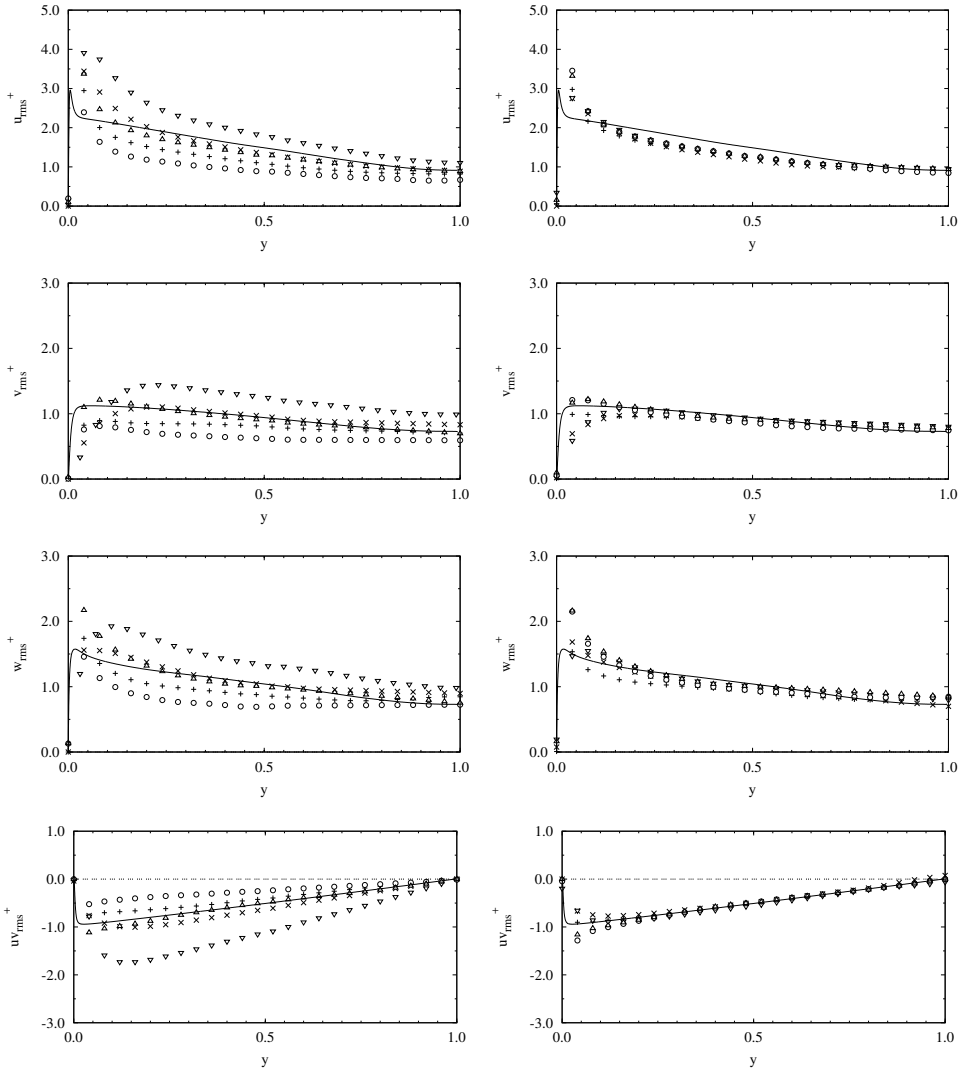

Figure 3.32: rms of the streamwise (u^+_{rms}), wall-normal (v^+_{rms}) and spanwise (w^+_{rms}) velocity fluctuations as well as the Reynolds shear stress (uv^+_{rms}) in wall units. The results have been obtained with five different LES models, DSMG (+), QR (×), SMG (▽), WALE (○), VMS-WALE (△) and two different wall strategies: LES-only (left) and WMLES at $h^+_{wm} \approx 30$ (right). DNS Ahn *et al.*[32] (————).

As expected, the LES-only plots show a large dispersion of results depending on the subgrid model and its particular near-wall formulation. Nonetheless, these dispersions significantly decrease when applying the WM, and the WMLES solutions seem to collapse fairly well onto the DNS curve regardless of the subgrid model used. This is especially true for the mean velocity profiles in the whole domain and the rms fluctuations away from the wall, while more dispersion is observed in the near-wall region for the latter. This is consistent with the fact that the WM acts as an integral part of the LES model and therefore standardizes the wall behavior for any subgrid formulation. Actually, providing a wall shear stress directly to the diffusive term is equivalent to provide a specific subgrid viscosity to the wall node such that the LES velocity gradient multiplied by the summation of the molecular and subgrid viscosities matches the τ_w computed by the model. This behavior is less evident when comparing velocity fluctuations near the wall. This is due to the necessity of having a minimum amount of resolved scales to capture them properly. Since the size of the eddies in the near-wall region is proportional to the wall distance and, therefore, very small compared with those in the pipe center, the mesh is unable to capture a minimum amount of near-wall structures when using coarse meshes with a constant grid spacing. According to the results, it seems clear that the WM is not able to redress it.

3.6 Conclusions

A general and efficient WM for incompressible LES and suitable for unstructured meshes is presented and validated through different tests including strongly unsteady non-equilibrium flows. The present methodology is included in the two-layer model family of wall shear stress models[7], and to the authors' best knowledge, it uses for the first time the full incompressible URANS equations as a mathematical and physical model. This mathematical approach has been selected given the importance of the non-equilibrium terms (i.e., advective and pressure gradient) when dealing with complex flow phenomena such as boundary layer detachment, adverse pressure gradients or flow recirculations.

Wall shear stress models in general and RANS-based WMs in particular, are affected by the "log-layer mismatch" and the resolved Reynolds stresses inflow problems which undermine the quality of the WM numerical predictions.

In the present work, a time-averaging filter for the LES variables is applied in the WM/LES interface to tackle both issues at once with a single-step and low-computational-cost technique suitable for any geometry. While the TAF strategy has already been applied to address the LLM problem[10], it is used for the first time to block the RRS inflow in the TLM context. According to the obtained results, this approach is extremely efficient in avoiding the RRS inflow consequences, and

combined with its effects on the LLM, it dramatically reduces the complexity of the WM formulation and implementation while strongly increasing its efficiency and geometrical range of applicability compared to existing TLM strategies[9].

The proposed methodology has been validated with two equilibrium Pipe Flow tests at $Re_\tau \approx 500$ and $Re_\tau \approx 3000$, and two non-equilibrium flows, specifically, a flow around a DU 91-W2-250 airfoil at $Re = 3 \times 10^6$ in full stall and a flow past an Ahmed car body at $Re = 7.56 \times 10^5$. As recurrently observed in all these tests, the signal of the LES variables must be filtered with a sufficiently large temporal filter length before being used as boundary condition for the WM RANS-based domain. Specifically, frequencies higher than the energy-containing/inertial range limit must be suppressed. While the apparent diffusive effects inherent to the resolved smallest scales are almost completely avoided with smaller filter lengths, it is found that the RANS model provides excessive diffusivity when inertial subrange frequencies are introduced through the boundary. The URANS approach exclusively resolves the mean flow evolution or the very-large structure unsteady component at most. Thus, for any RANS formulation, the turbulent viscosity should be derived from the mean flow, which is supposedly the only available data. When time-resolved turbulent motions are introduced through the boundary, they cause the RANS model to work out of its range of applicability, causing a prediction failure.

On the other hand, unlike in equilibrium conditions, for non-equilibrium unsteady flows, it also exists an upper bound for the TAF filter length. It is found that the largest flow structures contribution to the instantaneous skin friction plays an important role in regions with a strong unsteady behavior. Therefore, a sufficiently small filter length allowing to capture large-scale temporal effects must be used. In the particular case of the flow around a DU 91-W2-250 airfoil, a TAF averaging period corresponding approximately to a quarter of the largest flow time-scale allowed to obtain accurate predictions.

Additionally, it is found that the filtering period affects the large-scale characteristic frequencies. However, it remains unclear whether the value of T directly influences the largest time-scales, or this is an indirect consequence of having a poorly resolved mean flow.

Significant improvements in numerical results are obtained in all tests when using the WM. Nonetheless, the improving potential of the WM strongly depends on the characteristic flow physics of each case, especially on the boundary layer thickness and the Reynolds number. The first magnitude constrains the height at which the WM/LES interface can be placed and, therefore, the allowed coarsening of the LES mesh at the wall. The Reynolds number strongly conditions the gains that can be obtained from the time-step size. The thicker the boundary layer and the higher the Reynolds number, the larger the WM benefits will be.

On the other hand, the interaction between the wall and the subgrid model has

been analyzed by performing the WMLES pipe flow test at $(Re_\tau \approx 3000)$ with five different LES formulations. It was found that the WM acts as an integral part of the LES models, standardizing their near-wall behavior and making the disperse results obtained in the different LES-only tests to collapse into very similar solutions, reasonably close to the DNS reference.

Finally, another important conclusion of the present work is that the increase of the time-step upper bound in explicit computations has a huge potential in cutting down the computational costs for high Reynolds number flows. Probably, even more than the reduction of LES near-wall spatial resolution requirements since it is very difficult to parallelize the advancement in time. In order to support this idea, a new set of expressions for the total computational cost Reynolds number scaling which takes into account the time integration efforts is proposed. Further research on methodologies intended to mitigate the time-step effects are part of our research plans.

Bibliography

[1] J. Larsson, S. Kawai, J. Bodart, and I. Bermejo-Moreno. Large eddy simulation with modeled wall-stress: recent progress and future directions. *B. JSME*, 3:1–23, 2016.

[2] L. Prandlt. Bericht über die Entstehung der Turbulenz. *Angew. Math. Mech.*, 5:136–139, 1925.

[3] T. Von Kármán. Mechanishe Ähnlichkeit und Turbulenz. *Nachrichten von der Gesellschaft der Wissenshaften zu Göttingen, Fachgruppe 1 (Mathematik).*, 5:58–76, 1930.

[4] H. Schlichting and K. Gersten. *Boundary Layer Theory. 8th ed.* Springer-Verlag., 2004.

[5] J. Calafell, F.X. Trias, O. Lehmkuhl, and A. Oliva. A time-average filtering technique to improve the efficiency of Two-Layer Wall Models for Large Eddy Simulation in complex geometries. *Comput. Fluids*, 2019.

[6] E. Balaras and C. Benocci. Subgrid-scale models in finite-difference simulations of complex wall bounded flows. In *AGARD CP 551*, pages 2.1–2.5, 1994.

[7] E. Balaras, C. Benocci, and U. Piomelli. Two-layer approximate boundary conditions for large-eddy simulations. *AIAA J.*, 34 (6):1111–1119, 1996.

[8] S. Kawai and J. Larsson. Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy. *Phys. Fluids*, 24:015105, 2012.

[9] G. I. Park and P. Moin. An improved dynamic non-equilibrium wall-model for large eddy simulation. *Phys. Fluids*, 26:015108, 2014.

[10] X. I. A. Yang, G. I. Park, and P. Moin. Log-layer mismatch and modeling of the fluctuating wall stress in wall-modeled large-eddy simulations. *Phys. Rev. Fluids*, 2:104601, 2017.

[11] S. B. Pope. *Turbulent flows.* Cambridge Univesity Press, 2000.

[12] W. Cabot. Near-wall models in large-eddy simulations of flow behind a backward-facing step. *Annual Research Brief. Center for Turbulence Research, Stanford, CA*, pages 199–210, 1996.

[13] W. Cabot and P. Moin. Approximate wall boundary conditions in the large-eddy simulation of high Reynolds number flow. *Flow Turbul. Combust.*, 63:269–291, 1999.

[14] S. Kawai and J. Larsson. A dynamic wall model for large-eddy simulations of high Reynolds number compressible flows. *Annual Research Brief. Center for Turbulence Research, Stanford, CA*, pages 25–37, 2010.

[15] S. Kawai and K. Asada. Wall-modeled large-eddy simulation of high Reynolds number flow around an airfoil near stall condition. *Comput. Fluids*, 85:105–113, 2013.

[16] S. Kawai and J. Larsson. Dynamic non-equilibrium wall-modeling for large eddy simulation at high Reynolds number. *Phys. Fluids*, 25:015105, 2013.

[17] J. Larsson, S. Kawai, J. Bodart, and I. Bermejo-Moreno. Large-eddy simulation with modeled wall-stress: recent progress and future directions. *Mech. Engng. Rev.*, 3(1):1–23, 2016.

[18] M. Germano, U. Piomelli, P. Moin, and W. Cabot. A dynamic subgrid-scale eddy viscosity model. *Phys. Fluids*, 3:1760–1765, 1991.

[19] P. Moin, K. Squires, W. Cabot, and S. Lee. A dynamic subgrid-scale model for compressible turbulence and scalar transport. *Phys. Fluids*, 3:2746, 1991.

[20] D. K. Lilly. A proposed modification of the Germano subgrid-scale closure method. *Phys. Fluids*, 4:633–635, 1992.

[21] D. E. Aljure, O. Lehmkuhl, I. Rodríguez, and A. Oliva. Flow and turbulent structures around simplified car models. *Comput. Fluids*, 96:122–135, 2014.

[22] F. Nicoud and F. Ducros. Subgrid-scale stress modelling based on the square of the velocity gradient tensor. *Flow. Turbul. combust.*, 62:183–200, 1999.

[23] L. Jofre, O. Lehmkuhl, J. Ventosa, F. X. Trias, and A. Oliva. Conservation properties of unstructured finite-volume mesh schemes for the Navier-Stokes equations. *Numer. Heat Tr. B-Fund.*, 65:53–79, 2014.

[24] F. X. Trias, O. Lehmkuhl, A. Oliva, C. D. Pérez-Segarra, and R. Verstappen. Symmetry-preserving discretization of Navier-Stokes equations on collocated unstructured grids. *J. Comput. Phys.*, 258:246–267, 2014.

[25] R. W. C. P. Verstappen and A. E. P. Veldman. Symmetry-preserving discretization of turbulent flow. *J. Comput. Phys.*, 187:343–368, 2003.

[26] F. X. Trias and O. Lehmkuhl. A self-adaptive strategy for the time-integration of Navier-Stokes equations. *Numer. Heat Tr. B-Fund.*, 60-2:116–134, 2011.

[27] X. I. A. Yang, J. Sadique, R. Mittal, and C. Meneveau. Integral wall model for large eddy simulations of wall-bounded turbulent flows. *Phys. Fluids*, 27:025112, 2015.

[28] A. Carmona, O. Lehmkuhl, C. D. Pérez-Segarra, and A. Oliva. Numerical analysis of the transpose diffusive term for viscoplastic-type non-newtonian fluid flows using a collocated variable arrangement. *Numer. Heat Tr. B-Fund.*, 67:410–436, 2015.

[29] G. I. Park and P. Moin. Numerical aspects and implementation of a two-layer zonal wall model for LES of compressible turbulent flows on unstructured meshes. *J. Comput. Phys.*, 305:589–603, 2016.

[30] C. Chin, A. S. H. Ooi, I. Marusic, and H. M. Blackburn. The influence of pipe length on turbulence statistics computed from direct numerical simulation data. *Phys. Fluids*, 22:115107, 2010.

[31] S. Lardeau and M. Leschziner. Unsteady Reynolds-Averaged Navier-Stokes computations of transitional wake/blade interaction. *AIAA J.*, 42(8):1559–1571, 2004.

[32] J. Ahn, J. H. Lee, J. Lee, J. H. Kang, and H. J. Sung. Direct numerical simulation of a 30R long turbulent pipe flow at $\mathrm{Re}_\tau = 3008$. *Phys. Fluids*, 27:065110, 2015.

[33] W. A. Timmer and R. P. J. O. M. Van Rooij. Summary of the Delft university wind turbine dedicated airfoils. *J. Sol. Energ.*, 125:1–11, 2004.

[34] A. Lozano-Durán and J. Jiménez. Effect of the computational domain on direct simulations of turbulent channels up to $\mathrm{Re}_\tau = 4200$. *Phys. Fluids*, 26:011702–1, 2014.

[35] U. Piomelli and E. Balaras. Wall-layer models for large-eddy simulations. *Annu. Rev. Fluid Mech.*, 34:349–374, 2002.

[36] I. Rodríguez, O. Lehmkuhl, R. Borrell, and A. Oliva. Direct numerical simulation of a NACA0012 in full stall. *Int. J. Heat Fluid Flow*, 43:194–203, 2013.

[37] C. P. Mellen, J. Fröhlich, and W. Rodi. Lessons form LESFOIL project on large-eddy simulation of flow around an airfoil. *AIAA J.*, 41-4:573–581, 2003.

[38] K. Asada and S. Kawai. Large-eddy simulation of airfoil flow near stall condition at Reynolds number 2.1×10^6. *Phys. Fluids*, 30:085103, 2018.

[39] S. Pope. Ten questions concerning the large-eddy simulations of turbulent flows. *New J. Phys.*, 6(35):1–24, 2004.

[40] H. Lienhart and S. Becker. Flow and turbulence structure in the wake of a simplified car model. In *SAE Technical paper*. SAE International, 2003.

[41] J. Smagorinsky. General circulation experiments with the primitive equations, I. the basic experiment. *Monthly Weather Review*, 91:99–164, 1963.

[42] A. A. Wray and J. C. R. Hunt. Algorithms for classification of turbulent structures. In *Topological Fluid Mechanics, Proceedings of the IUTAM Symposium.*, number 95–104, 1989.

[43] T. J. R. Hughes, L. Mazzei, and K. E. Jansen. Large-eddy simulation and the variational multiscale method. *Comput. Vis. Sci.*, 3:47–59, 2000.

[44] L. Bricteuxa, M. Duponcheel, and G. Winckelmans. A multiscale subgrid model for both free vortex flows and wall-bounded flows. *Phys. Fluids*, 25:105102, 2009.

[45] O. Lehmkuhl, I. Rodríguez, A. Báez, A. Oliva, and C. D. Pérez-Segarra. On the large-eddy simulations for the flow around aerodynamic profiles using unstructured grids. *Comput. Fluids*, 84:176–189, 2013.

[46] R. Verstappen. When does eddy viscosity damp subfilter scales sufficiently? *J. Sci. Comput.*, 49:94–110, 2011.

4

Mathematical and numerical formulation of the Two-Layer wall model

In the present chapter, the numerical strategy used to solve the TLM physical and mathematical model of Chapter 3, will be developed and discussed in detail. Initially, a short introduction about the Finite Volume Method (FVM) is given. In parallel, the concept of property conservation of continuous operators in the discrete domain is introduced. The purpose of this brief overview, is to understand better the mathematical developments carried along the Chapter, as well as, to show the philosophy behind the FVM and the operator property conservation approach when it comes to the resolution of a physical model. Then, the FVM will be applied to the continuous mathematical model in order to obtain a numerical frame allowing us to obtain discrete solutions for the TLM governing equation. At the end of the Chapter, the global numerical procedure is summarized with a flow chart.

4.1 Numerical resolution of the model governing equations.

As previously introduced in Chapter 1, in general, there are no analytical solutions for PDEs, and the N-S equations are not an exception. Hence, a numerical method has to be used in order to obtain discrete solutions of these equations. Since the physical phenomenon described by the N-S equations is time and space-dependent, a suitable discretization must be applied for all dimensions.

There are different numerical techniques to perform the spatial discretization, such as finite differences, finite volumes or finite elements. All of them are intended to discretely represent a given PDE and its spatial domain through a finite set of algebraic equations, each one representing a specific spatial position in which a numerical solution of the original PDE can be found. Each methodology mentioned above has its strengths and weaknesses, being all these techniques widely used in CFD, which indicates that none of them offers overwhelmingly clear advantages over the others.

TermoFluids is a finite volume-based code. The main strength of this approach is that it is relatively easy to enforce a physic compatible numerical behavior through the discretization schemes. That means that the numerical solution of the equations conserves the physical properties of their continuous counterparts[1]. For instance, the continuous solution of the Navier-Stokes equations conserves the kinetic energy balance. The discrete solution does not automatically hold this property, and suitable numerical schemes for both, the temporal and the spatial discretization, have to be used to ensure the conservation properties in the discrete domain[1, 2, 3, 4]. The reproduction of the continuous physical model properties in the discrete domain is a crucial aspect to obtain realistic flow simulations, and at the same time, it contributes to the numerical stability of the algorithm without needing to resort to unphysical methodologies to ensure the computation robustness.

The main purpose of the initial sections is to briefly introduce the finite volume method as a technique to numerically resolve a set of partial differential equations (PDE). The objective is not to provide an extensive review of numerical methods, but to make a short introduction to follow the discretization process of the RANS equations easily. On the other hand, the continuous/discrete operator property conservation concept is also introduced.

4.2 The finite volume method.

The finite volume method (FVM) is a mathematical approach to transform a set of partial differential equations into a finite set of algebraic equations [5, 6] in order to obtain a discrete numerical solution of the original PDEs. The discrete solution is obtained in a finite set of points called nodes. These nodes are distributed throughout the computational domain in which the solution has to be obtained. The "finite volume" refers to the small volume surrounding each node point. The continuous solution domain is subdivided in finite volumes or cells, forming the so-called computational mesh which includes the node points at the volume's centroid and at the volume's faces that belong to the domain boundary. The higher the number of cells, the higher will be the resolution of the numerical solution. Unfortunately, for most PDEs, especially for high non-linear ones, the numerical solution is strongly

dependent on the mesh resolution for the reasons given in Chapter 1.

In FVM, the PDE is integrated in the finite control volume considering the value of all variables constant throughout the integration domain. This value, is the average of the PDE variable ϕ within a cell of volume Ω:

$$\overline{\phi} = \frac{1}{\Omega} \oint_{\Omega} \phi \, d\Omega, \tag{4.1}$$

On the other hand, in a PDE containing a divergence term, the volume integrals are converted to surface integrals by using the divergence or Gauss theorem. These terms are then evaluated as fluxes at the surfaces of each finite volume. Analogously to the volumetric average (Equation 4.1), the value of ϕ is considered constant throughout a given cell face. Since the solution values of ϕ are known at the volume centroids (nodes) instead of at the surfaces, it is necessary an interpolation method or spatial numerical scheme to evaluate the fluxes at the volume faces.

The mathematical statement for the divergence theorem in a $\mathbb{R}^3$ space is as follows: suppose Ω a three-dimensional volume which is compact and has a piecewise and smooth boundary S ($\partial\Omega = S$). If $\mathbf{f}$ is a continuously differentiable vector field defined in a neighborhood of Ω, then we have:

$$\oint_{\Omega} (\nabla \cdot \mathbf{f}) \, d\Omega = \oint_{S} (\mathbf{f} \cdot \mathbf{n}) \, dS, \tag{4.2}$$

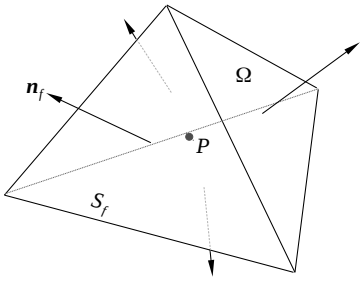

Figure 4.1: Discrete cell of volume Ω. P is the cell centroid (node) while $\boldsymbol{n}_f$ and S_f, are the cell's face normal vector and surface, respectively.

Below, an example of a PDE discretization through FVM is given. For the sake of simplicity and to easily understand the discretization process, the relatively simple Laplacian PDE is considered:

$$\nabla^2 \phi = \nabla \cdot \nabla \phi = B, \tag{4.3}$$

being ϕ a given scalar field defined in a three-dimensonal space ($\phi : \mathbb{R}^3 \rightarrow \mathbb{R}$) and B, a scalar source term $B = B(x, y, z)$. For a particular control volume Ω (see figure 4.1), the PDE can be integrated over its domain:

$$\oint_{\Omega} \nabla \cdot \nabla \phi \, d\Omega = \oint_{\Omega} B \, d\Omega, \tag{4.4}$$

Afterward, the Gauss theorem (Equation 4.2) is applied to the divergence term such that Equation 4.4 can be rewritten as:

$$\oint_{\Omega} \nabla \cdot \nabla \phi \, d\Omega = \oint_{S} \nabla \phi \cdot \mathbf{n} \, dS = \oint_{\Omega} B \, d\Omega, \tag{4.5}$$

In general, the spatial discretization of the computational domain is performed through polyhedrons of n faces. According to the hypothesis of constant ϕ throughout the cell volume and at each particular face, equation 4.6 can be rewritten in terms of space-averaged values and a summation of fluxes over a finite number of faces:

$$\sum_{f=1}^{m} \nabla \phi_f \cdot \mathbf{n}_f S_f = B_P \, \Omega, \tag{4.6}$$

where B_P is the value of the source term at point P, $\nabla \phi_f$, is the gradient of ϕ at a given face, $\boldsymbol{n}_f$ is the face's normal vector, and m, is the total number of cell faces. In order to obtain an algebraic equation for ϕ, the value of the scalar field gradient at the cell faces has to be evaluated. There are different ways to determine $\nabla \phi_f$. However, not all of them would preserve the properties of the continuous Laplacian operator in the discrete form.

The continuous Laplacian operator is symmetric and negative definite. Mathematically, these two properties can be respectively expressed as follows prodided that there are no boundary contributions:

$$\int_{\Omega} f \, (\nabla \cdot \nabla g) \, d\Omega = \int_{\Omega} (\nabla \cdot \nabla f) \, g \, d\Omega, \tag{4.7}$$

$$\int_{\Omega} f \, \nabla \cdot \nabla f \, d\Omega = - \int_{\Omega} |\nabla f|^2 \, d\Omega < 0, \tag{4.8}$$

where f and g are two functions such that $f, g : \mathbb{R}^3 \rightarrow \mathbb{R}$.

Hereafter, the discrete Laplacian operator for the simple spatial discretization (mesh) in Figure 4.2 will be obtained from Equation 4.6. A central-difference symmetry preserving scheme will be used to evaluate the ϕ gradient at the cell faces. The projection of $\nabla \phi_f$ in the face-normal direction can be approximated as follows:

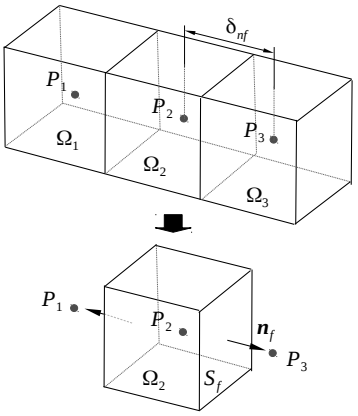

Figure 4.2: Example of small computational mesh, composed of three cells. δ_{nf} is the face-normal distance between nodes.

$$\nabla\phi_f \cdot \mathbf{n}_f \approx \frac{\phi_{NB} - \phi_P}{\delta_{n_f}}, \tag{4.9}$$

where the subindex NB stands for the cell neighbor node contiguous to face f, and δ_{n_f}, is the distance between node NB and node P in the face normal direction $(\mathbf{n}_f)$. For the sake of clarity, a uniform mesh with equal geometric parameters is considered (distances δ_{n_f}, surfaces S_f and volumes Ω_i), although the conclusions would be the same if a fully unstructured mesh was used. Consequently, the discrete equations for nodes 1,2 and 3 are:

$$\begin{aligned}
1: &\quad \frac{\phi_{P_2} - \phi_{P_1}}{\delta_{n_f}} S &&= B_{P_1}\Omega_1 \;\rightarrow\; -\phi_{P_1} + \phi_{P_2} &&= B_{P_1}\frac{\delta_{n_f}\Omega}{S}, \\
2: &\quad \frac{\phi_{P_1} - \phi_{P_2}}{\delta_{n_f}} S + \frac{\phi_{P_3} - \phi_{P_2}}{\delta_{n_f}} S = B_{P_2}\Omega \;\rightarrow\; &&\phi_{P_1} - 2\phi_{P_2} + \phi_{P_3} &&= B_{P_2}\frac{\delta_{n_f}\Omega}{S}, \\
3: &\quad \frac{\phi_{P_2} - \phi_{P_3}}{\delta_{n_f}} S &&= B_{P_3}\Omega \;\rightarrow\; &&\phi_{P_2} - \phi_{P_3} = B_{P_3}\frac{\delta_{n_f}\Omega}{S},
\end{aligned} \tag{4.10}$$

Rearranging Equation set 4.10 in matrix form, we obtain:

$$\frac{S}{\delta_{n_f}} \begin{bmatrix} -1 & 1 & 0 \\ 1 & -2 & 1 \\ 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} \phi_{P_1} \\ \phi_{P_2} \\ \phi_{P_3} \end{bmatrix} = \begin{bmatrix} B_{P_1}\Omega \\ B_{P_2}\Omega \\ B_{P_3}\Omega \end{bmatrix} \quad \Rightarrow \quad L\phi_P = B_P\Omega, \tag{4.11}$$

Comparing the continuous Equation 4.3 with its discrete approximation 4.11, it can be noted that the discrete Laplacian operator L is symmetric and negative definite as its continuous counterpart ∇^2. For the discrete operator, these two attributes can be mathematically defined as $L = L^T$ and $(\mathbf{x}^T \cdot L\mathbf{x}) < 0;\ \forall\mathbf{x} \in \mathbb{R}^n,\ \mathbf{x} \neq 0$ [7], respectively. This continuous/discrete analogy is not automatically fulfilled, and appropriate numerical schemes, such as the central-difference used in the previous example, have to be used. In Section 4.3, the discretization of the RANS equations will be carried out, and more details on conservative numerical schemes will be given.

Another aspect that has to be considered when applying a given numerical scheme is its order of accuracy. Higher-order methods use more points to interpolate the variables at a given point than lower-order ones. The use of high-order numerical methods reduces truncation errors denoted as $\mathcal{O}(h^{n+1})$, where h is the discretization step size of the independent variable (usually, space or time in physical models), and n is order of accuracy [8].

Since in FVM the discrete problem variables are defined at a given cell inner point (nodes), usually the cell centroid, the face values have to be interpolated. Nonetheless,

one of the major drawbacks in the FVM approach resides in the difficulty of applying high order schemes in unstructured meshes, which is almost unavoidable when dealing with complex geometries. This problem is derived from the difficulty of determining the face neighboring nodes. While the application of a second-order scheme is trivial (the two face adjacent nodes), the following points needed for higher-order schemes might not be evident at all.

4.3 Numerical resolution of the URANS equations.

The non-equibrium TLM mathematical model presented in Chapter 3 is based on the incompressible Unsteady Reynolds-Averaged Navier-Stokes equations:

$$\nabla \cdot \langle \mathbf{u} \rangle = 0, \tag{4.12}$$

$$\frac{\partial \langle \mathbf{u} \rangle}{\partial t} + \left(\langle \mathbf{u} \rangle \cdot \nabla \right) \langle \mathbf{u} \rangle = \nabla \cdot \left[2(\nu + \nu_{Twm}) S(\langle \mathbf{u} \rangle) \right] - \nabla \langle p \rangle, \tag{4.13}$$

being $\langle \boldsymbol{u} \rangle$, the time-averaged velocity field, ν, the kinematic viscosity, S, the rate-of-strain tensor, and $\langle p \rangle$, the time-averaged kinematic pressure. To close the formulation, the value of ν_{Twm} has to be provided through an external model. As previously commented, a mixing-length-based algebraic model together with the Van Driest wall-damping function is used for the present implementation:

$$\nu_{Twm} = (\kappa y)^2 |S| \left[1 - exp\left(-y^+ / A^+ \right) \right]^2, \tag{4.14}$$

where $\kappa = 0.41$, is the von Kármán constant, y, is the wall distance, $|S|$, is the magnitude of the rate-of-strain tensor, $A^+ = 26$, is a constant of the wall-damping function, and the superindex $+$, denotes wall units.

In order to follow a consistent criterion all along the thesis document, the operator $\langle \cdot \rangle$ has been used to denote time-averaged variables. However, since in the following sections all the problem variables will be time-averaged, the $\langle \cdot \rangle$ symbol will be obviated for the sake of clarity.

The RANS equations are far more complex than the Laplacian example presented in the previous section. They are a set of four scalar PDEs in which the problem variables are the three velocity components ($\boldsymbol{u} = (u, v, w)$) and the pressure p. The scalar equations are coupled through the velocity and the pressure, and therefore, a mathematical methodology has to be applied to solve the velocity-pressure coupling. Additionally, the equations feature a non-linear spatial term ($(\boldsymbol{u} \cdot \nabla) \boldsymbol{u}$), being necessary a linearization step to obtain a system of linear algebraic equations. Finally, the equations feature a transient term, ($\partial \boldsymbol{u} / \partial t$), for which a suitable temporal discretization scheme has to be decided.

4.3.1 Velocity-pressure coupling: The Fractional-step method

Several methodologies are allowing to obtain a numerical solution for the RANS equations. In the present work, a projection method, introduced by Chorin in 1968 [9] is used. The main advantage of this method is that the computations of the velocity and the pressure fields are decoupled, and can be evaluated in a succession of steps. This leads to better performance and simplicity of the algorithm compared to other existing methodologies like SIMPLE [10, 11].

This method is based on the Helmholtz decomposition [12, 13] (See Appendix A for more details). The theorem states that a given vector field $\boldsymbol{u}$ defined on a simply connected domain can be uniquely decomposed into two orthogonal components, a divergence-free (solenoidal) part $\boldsymbol{u}_{sol}$, and an irrotational field $\boldsymbol{u}_{irr}$. On the other hand, for any scalar function ϕ, its gradient is irrotational,i.e., the curl of the gradient is zero. Thus,

$$\boldsymbol{u} = \boldsymbol{u}_{sol} + \boldsymbol{u}_{irr} \xrightarrow{\nabla\times\nabla\phi=0} \boldsymbol{u} = \boldsymbol{u}_{sol} + \nabla\phi, \tag{4.15}$$

This mathematical property will allow us to decompose the incompressible RANS equations into two uncoupled parts while making clear the role of the pressure on the global algorithm. Let $\Pi(\cdot)$ be a projector operator which projects any vector field $\boldsymbol{u}$ onto a divergence-free space:

$$\nabla \cdot \Pi(\boldsymbol{u}) = 0, \tag{4.16}$$

Reordering the incompressible RANS equations (4.13) and applying Π, we obtain:

$$\Pi\left[\frac{\partial \boldsymbol{u}}{\partial t} + \nabla p\right] = \Pi\left[\left[-\left(\boldsymbol{u}\cdot\nabla\right)\boldsymbol{u}\right] + \nabla\cdot\left[2\nu_{eff}S(\boldsymbol{u})\right]\right], \tag{4.17}$$

where ν_{eff} stands for $\nu_{eff} = \nu + \nu_{Twm}$. Since the velocity field $\boldsymbol{u}$ is incompressible, i.e. it is solenoidal (See equation 4.12), it remains unchanged when projected onto the divergence-free space, and therefore, its transient term too.

$$\Pi\left(\frac{\partial \boldsymbol{u}}{\partial t}\right) = \frac{\partial \boldsymbol{u}}{\partial t}, \tag{4.18}$$

On the other hand, according to the Helmholtz-Hodge decomposition, the pressure gradient is orthogonal to the divergence-free space, and therefore, its projection is zero:

$$\Pi\left(\nabla p\right) = 0, \tag{4.19}$$

Hereafter, Equation 4.13 and Equation 4.17 will be compared, considering the two previous statements:

$$\frac{\partial \boldsymbol{u}}{\partial t} = \quad \left[-\left(\boldsymbol{u}\cdot\nabla\right)\boldsymbol{u} + \nabla\cdot\left[2\nu_{eff}S(\boldsymbol{u})\right]\right] - \nabla p, \tag{4.20}$$

$$\downarrow$$

$$\frac{\partial \boldsymbol{u}}{\partial t} = \Pi\left[-\left(\boldsymbol{u}\cdot\nabla\right)\boldsymbol{u} + \nabla\cdot\left[2\nu_{eff}S(\boldsymbol{u})\right]\right], \tag{4.21}$$

From this comparison, it can be observed that the role of the pressure gradient is to project the convective and the diffusive terms of the RANS equations onto the divergence-free space. This property will be used to compute the velocity field $\boldsymbol{u}$ and the pressure p in two uncoupled steps. First of all, the predictor velocity, $\boldsymbol{u}^P$, will be defined as the non-projected velocity field ($\boldsymbol{u} = \Pi(\boldsymbol{u}^P)$). This definition also applies to the temporal derivative of the velocity field:

$$\frac{\partial \boldsymbol{u}}{\partial t} = \Pi\left[\frac{\partial \boldsymbol{u}^P}{\partial t}\right], \tag{4.22}$$

Then, according to Equation 4.21, the temporal derivative of $\boldsymbol{u}^P$ corresponds to the inner part of the projection operator:

$$\frac{\partial \boldsymbol{u}^p}{\partial t} = -\left(\boldsymbol{u}\cdot\nabla\right)\boldsymbol{u} + \nabla\cdot\left[2\nu_{eff}S(\boldsymbol{u})\right], \tag{4.23}$$

On the other hand, combining Equations 4.20 and 4.23, we obtain:

$$\frac{\partial \boldsymbol{u}}{\partial t} = \frac{\partial \boldsymbol{u}^P}{\partial t} - \nabla p, \tag{4.24}$$

Then, approximating the temporal derivatives with a second-order central difference numerical scheme and reordering terms, we obtain:

$$\frac{\boldsymbol{u}^{n+1} - \boldsymbol{u}^n}{\Delta t} = \frac{\boldsymbol{u}^P - \boldsymbol{u}^n}{\Delta t} - \nabla p \qquad \longrightarrow \qquad \boldsymbol{u}^{n+1} = \boldsymbol{u}^P - \Delta t \nabla p, \tag{4.25}$$

where the superindex n defines the discrete temporal point. The obtained expression corresponds to the Helmholtz-Hodge decomposition (see Equation 4.15). in which the scalar function whose gradient projects the velocity field on the divergence-free space corresponds to the pseudo-pressure $\tilde{p}$, defined as $\tilde{p} = \Delta t\, p$. The unicity of the decomposition ensures that for a given pseudo-pressure field, there is only a solenoidal velocity field that fulfills the following equation:

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^p - \nabla \tilde{p}, \tag{4.26}$$

However, the pseudo-pressure field still remains unknown, and prior to apply this step, it has to be determined. Applying the divergence operator to Equation 4.26, we obtain a Poisson equation for the pressure:

$$\nabla \cdot \boldsymbol{u} = \nabla \cdot \boldsymbol{u}^p - \nabla \cdot \nabla \tilde{p} \xrightarrow{\nabla \cdot \boldsymbol{u}=0} \nabla^2 \tilde{p} = \nabla \cdot u^p, \tag{4.27}$$

In general, Equation 4.27 is indeterminate and some additonal information has to be provided to obtain a specific solution. Specifically, for the discrete solution, one of the main diagonal coefficients is artificially increased to condition the equation system. Therefore, the mathematical procedure to uncouple the velocity and the pressure fields allowing to obtain them separately, can be summarized as follows:

$$\frac{\partial \boldsymbol{u}^p}{\partial t} = -\left(\boldsymbol{u} \cdot \nabla\right) \boldsymbol{u} + \nabla \cdot \left[2\nu_{eff} S(\boldsymbol{u})\right] \quad \text{(Eq. 4.23)}$$

$$\downarrow$$

$$\nabla^2 \tilde{p} = \nabla \cdot \boldsymbol{u}^p \quad \text{(Eq. 4.27)}$$

$$\downarrow$$

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^p - \nabla \tilde{p} \quad \text{(Eq. 4.26)}$$

Up to this point, a sequential algorithm allowing to determine $\boldsymbol{u}$ and p separately has been created. Now, a numerical methodology has to be applied to obtain a numerical solution for the procedure described above. The finite volume method will be used to discretize the spatial derivatives while the temporal term in Equation 4.23 will be approximated through a first-order Backward Euler Implicit method. In the present formulation, a collocated mesh scheme will be used. That means that all the problem variables will be numerically solved in the same grid points.

4.3.2 Spatial discretization of the predictor velocity equation

In this section, the FVM will be applied to equation 4.23 to spatially discretize the PDE. Since in this section ν_{eff} and u^p cannot be confused with other magnitudes, the superindex $()^p$ and the subindex $()_{eff}$ will be suppressed for the sake of clarity. On the other hand, given that the following development requires different equations for each equation component, a subindex notation will be used for the variables to obtain more compact expressions.

First of all, equation 4.23 will be rearranged by multiplying it by the density ρ and witting the convective term in conservative form:

$$\rho \frac{\partial u_i}{\partial t} = -\nabla \cdot \left(\rho u_i u_j\right) + \nabla \cdot \left[2\mu S_{ij}\right], \tag{4.28}$$

Then, the equation is integrated in a finite control volume Ω:

$$\oint_\Omega \rho \frac{\partial u_i}{\partial t} d\Omega = -\oint_\Omega \nabla \cdot \left(\rho u_i u_j\right) d\Omega + \oint_\Omega \nabla \cdot \left[2\mu S_{ij}\right] d\Omega, \tag{4.29}$$

Afterward, by applying the divergence theorem, we obtain:

$$\oint_\Omega \rho \frac{\partial u_i}{\partial t} d\Omega = -\oint_S \left(\rho u_i u_j\right) n_i dS + \oint_S 2\mu S_{ij} n_i dS, \tag{4.30}$$

Hereafter, a polyhedral finite volume (see Figure 4.1) is considered, and the FVM hypothesis of averaged values of the transported variables at the cell centroid and faces is applied (see Section 4.2). Therefore, the volume and surface integrals can be evaluated since the variable values throughout the integration domain remain constant:

$$\rho \frac{\partial u_{iP}}{\partial t} \Omega = -\sum_{f=1}^{m} \left(\rho u_{if} u_{jf}\right) n_{if} S_f + \sum_{f=1}^{m} 2\mu_f S_{ijf} n_{if} S_f, \tag{4.31}$$

where the subindex P and f denote the space-averaged value of a given magnitude at the cell and the polyhedron faces centroids, respectively. The convective and the diffusive terms have been transformed into a summation of fluxes at the cell faces while the variable values are known at the cell centroids. Therefore, interpolating numerical schemes have to be applied to evaluate them. Below, the particular discretization of each term will be detailed.

Convective term discretization

As commented above, the convective term has been transformed into a summation of fluxes at the cell faces, and therefore, the value of the variables have to be interpolated through a spatial numerical scheme from the cell node to the face centroid. On the other hand, the continuous convective operator $(u_i \cdot \nabla)$ is skew-symmetric, and this property has to be conserved by the discrete operator in order to keep the physical properties of the continuous model. Not all the interpolation schemes ensure the skew-symmetry of the matrix operator. According to Verstappen and Veldman, if the face value is computed as the averge of the face neighboring nodes values $(0.5(u_{iP} + u_{iNB})$ see Figure 4.3), the skew-symmetric property is preserved provided that the incompressibility constrain is fulfilled[1].

The convective term in equation 4.31 can be rewritten as follows since $\rho u_{jf} n_{if} S_f$ stands for the mass flow $\dot{m}$ through a given cell surface f:

$$\sum_{f=1}^{m} \left(\rho u_{if} u_{jf}\right) n_{if} S_f = \sum_{f=1}^{m} \dot{m}_f u_{if}, \tag{4.32}$$

Then, as commented above, the value of u_{if} will be approximated at the cell face through a second-order scheme representing one half of the sum of the u_i values at the face neighbor nodes (see Figure 4.3):

$$\sum_{f=1}^{m} \dot{m}_f u_{if} \approx \sum_{f=1}^{m} \dot{m}_f 0.5(u_{iNB} + u_{iP}), \tag{4.33}$$

where the subindex $()_{NB}$ stands for the value of a given variable at the face-contiguous cell node.

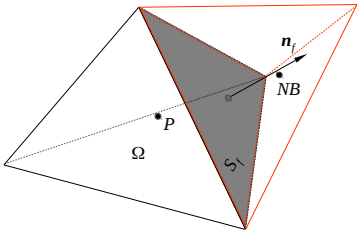

Figure 4.3: Pair of neighboring cells sharing a common face of surface S_f and normal vector $\boldsymbol{n}_f$. Main cell of node P in black, neighbor cell of node NB in orange.

Diffusive term discretization

S_{ij_f} stands for the rate-of-strain tensor ($S_{ij} = 1/2(\partial u_i/\partial x_j + \partial u_j/\partial x_i)$) evaluated at f. By applying the Gauss theorem, the diffusive term becomes a summation of velocity gradients at the cell faces:

$$\sum_{f=1}^{m} 2\mu_f S_{ij_f} n_{if} S_f = \sum_{f=1}^{m} \mu_f \left[2\frac{\partial u_i}{\partial x_i}\bigg|_f n_{if} + \left[\frac{\partial u_i}{\partial x_j}\bigg|_f + \frac{\partial u_j}{\partial x_i}\bigg|_f \right] n_{jf} \right] S_f \quad i \neq j, \tag{4.34}$$

In order to obtain an algebraic equation, all the spatial derivatives have to be approximated at the cell faces through a numerical scheme but preserving the properties of the original continuous diffusive operator which is symmetric. According Verstappen and Veldman, a central-difference scheme conserves the symmetry of the continuous operator:

$$\frac{\partial u_i}{\partial x_i}\bigg|_f \approx \frac{u_{iNB} - u_{iP}}{\delta_i}, \tag{4.35}$$

where δ_i is the distance between the two face-neighboring nodes, P and NB in direction i.

By applying Equation 4.35 to 4.34, the discretized components of the momentum equation become a set of linear combinations of discrete values of u_i at the mesh nodes,

in which the constant values only depend on geometric parameters and physical properties. For $i \neq j$ we obtain:

$$\sum_{f=1}^{m} 2\mu_f S_{ij_f} n_{if} S_f \approx \sum_{f=1}^{m} \mu_f \left[2n_i \frac{u_{iNB} - u_{iP}}{\delta_i} + n_j \left[\frac{u_{iNB} - u_{iP}}{\delta_j} + \frac{u_{j_{NB}} - u_{j_P}}{\delta_i} \right] \right] S_f, \tag{4.36}$$

4.3.3 Temporal integration of the predictor velocity equation

The only remaining continuous operator is the temporal derivative. Therefore, a temporal discretization scheme has to be applied to approximate the temporal derivative and to obtain an entirely algebraic equation in terms of discrete variables. Additionally, a time integration method has to be applied. There are two main different time integration schemes, explicit and implicit. For a Two-Layer Wall model, the main requirement of the temporal integration scheme is the numerical stability of the method regardless of the time step size. As explained in Chapter 1, the CFL limit sets the maximum allowed Δt size from which the temporal resolution is not sufficient to capture the flow structures supported by a given mesh resolution. Additionally, for explicit time integration schemes, it also sets a stability threshold. For time step sizes larger than the CFL limit, the computation becomes unstable. By contrast, implicit schemes, although losing physical information due to the poor temporal resolution, they are stable at any time step size.

The WM mesh has to be able to capture the steep velocity gradients near the wall. On the other hand, in WMLES, the smallest scales to be captured by the LES mesh, are those of the outer layer whose size is proportional to the wall distance, and therefore, of the same order of magnitude as the wall model total height. Therefore, in general, the WM mesh resolution will be much finer than its LES counterpart. This means that the most restrictive CFL value will probably come from the WM mesh, given that the CFL condition is highly dependent on the grid size. By applying an implicit scheme in the TLM formulation, this restrictive CFL limit is circumvented, allowing the use of the larger Δt of the LES domain, which is one of the main strengths of the TLM approach. In order to mitigate the errors caused by the use of large time steps, the RANS model is used to counterbalance the lost physical information.

In the present TLM implementation, a backward Euler scheme is used. Although this methodology is for any differential equation featuring a temporal derivative, including complex PDE such as the RANS equations, for the sake of clarity, the concept will initially be explained with a simple ordinary differential equation (ODE):

$$\frac{dy}{dt} = f(t, y), \tag{4.37}$$

where $y = y(t)$ is the unknown function, and f is a function of y and t. The temporal derivative term, is approximated as follows:

$$\left.\frac{dy}{dt}\right|_{n+1/2} \approx \frac{y^{n+1} - y^n}{\Delta t}, \tag{4.38}$$

being n the discrete temporal position along time. Since the values of y and t are only known in discrete temporal positions, the values of f will also be known on these positions. Which characterizes an implicit method, is that the values of f are defined at the $n+1$ position, entailing that they are unknown at the time of evaluating y^{n+1} and therefore, an algebraic equation has to be derived to determine the variable values at $n+1$.

$$\frac{y^{n+1} - y^n}{\Delta t} = f(y^{n+1}, t^{n+1}), \tag{4.39}$$

If we apply this approach to equation 4.31 we obtain:

$$\rho\Omega \frac{u_{i_P}^{n+1} - u_{i_P}^n}{\Delta t} \approx -\sum_{f=1}^{m} \left(\rho u_{i_f}^{n+1} u_{i_f}^{n+1}\right) n_{if} S_f + \sum_{f=1}^{m} 2\mu_f S_{ij_f}^{n+1} n_{if} S_f, \tag{4.40}$$

4.3.4 The algebraic equation of the predictor velocity

If we reconstruct equation 4.31 with the discretized approximations obtained in previous sections, we obtain algebraic equation for the predictor velocity:

$$\begin{aligned}\rho\Omega \frac{u_{i_P}^{n+1} - u_{i_P}^n}{\Delta t} = &-\sum_{f=1}^{m} \dot{m}_f 0.5(u_{i_{NB}}^{n+1} + u_{i_P}^{n+1}) + \\ &+ \sum_{f=1}^{m} \mu_f \left[2n_i \frac{u_{i_{NB}}^{n+1} - u_{i_P}^{n+1}}{\delta_i} + n_j \left[\frac{u_{i_{NB}}^{n+1} - u_{i_P}^{n+1}}{\delta_j} + \frac{u_{j_{NB}}^{n+1} - u_{j_P}^{n+1}}{\delta_i}\right]\right] S_f \quad i \neq j,\end{aligned} \tag{4.41}$$

Regrouping the discrete variables on the left, and the independent terms on the right, we obtain the following algebraic equation which relates the value of u_i at a given node P with the values u_i and u_j at its neighboring nodes NB:

$$a_{iP} u_{i_P}^{n+1} + \sum_{f=1}^{m} a_{iNB} u_{i_{NB}}^{n+1} + a_{jP} u_{j_P}^{n+1} + \sum_{f=1}^{m} a_{jNB} u_{j_{NB}}^{n+1} = b_{iP}, \tag{4.42}$$

Where:

$$a_{iP} = \frac{\rho\Omega}{\Delta t} + \sum_{f=1}^{m} 0.5\dot{m}_f + \sum_{f=1}^{m} \frac{2\mu_f n_{if} S_f}{\delta_i} + \sum_{f=1}^{m} \frac{\mu_f n_{jf} S_f}{\delta_j}, \tag{4.43}$$

$$a_{iNB} = 0.5\dot{m}_f - \frac{2\mu_f n_{if} S_f}{\delta_i} - \frac{\mu_f n_{jf} S_f}{\delta_j}, \tag{4.44}$$

$$a_{jP} = \sum_{f=1}^{m} \frac{\mu_f n_{jf} S_f}{\delta_i}, \tag{4.45}$$

$$a_{jNB} = -\frac{\mu_f n_{jf} S_f}{\delta_i}, \tag{4.46}$$

$$b_{iP} = \frac{\rho\Omega}{\Delta t} u_{iP}^{n}, \tag{4.47}$$

where $i \neq j$ for all coefficients. As can be observed, all of them depend on geometrical parameters, physical properties, and the mass flow $\dot{m}_f$ which is the contribution of the non-linear convective term. The mass flow depends on the velocity field itself causing a non-linearity in the algebraic equation that has to be properly addressed. To do so, an iterative process in which the velocity field is solved for the iteration $k+1$ by using the mass flow evaluated at iteration k, is applied (see Figure 4.4). This approach transforms the non-linear algebraic equation into a linear equation which can be easily solved through a standard linear solver.

Boundary conditions

In order to obtain a specific solution for a PDE, a set of boundary conditions for the equation must be given. In general, for parabolic and elliptic problems, two types of boundary conditions are considered, Dirichlet and Neumann. The first condition is a fixed value condition, in which the value of the problem variable at the boundary is known. On the other hand, the Neumann condition specifies the value of the variable gradient at the boundary. The mathematical expression of these conditions in discrete form is as follows:

Dirichlet condition:

$$u_{iP} = U_{iB} \longrightarrow a_{iP} u_{iP} = b_{iP} \quad ; \quad a_{iP} = 1,\, b_{iP} = U_{iB}, \tag{4.48}$$

where U_{iB} is the specified value of the variable u_i at the boundary node P.

Neumann condition:

$$\frac{u_{iNB} - u_{iP}}{\delta_n} = G_{iP} \longrightarrow a_{iNB}u_{iNB} - a_{iP}u_{iP} = b_{iP}\ ;\ a_{iP} = -1,\ a_{iNB} = 1,\ b_{iP} = \delta_n G_{iB}, \tag{4.49}$$

where G_{iB} is the specified value of the u_i gradient at the boundary, and δ_n is the normal distance between the boundary node and its neighbor inner node.

Therefore, the set of discrete equations for $\boldsymbol{u}^P$ will have one equation of the 4.3.4 type for each inner node, and one equation of the 4.48 or 4.49 type for each boundary node.

4.3.5 Poisson equation spatial discretization

Once the predictor velocity has been obtained at $n+1$, the pressure is needed in order to project u^P onto the divergence-free space (see Equation 4.26). Therefore, the Poisson PDE equation for the pressure (4.27) has to be numerically solved. As in previous steps, the finite volume method is applied:

$$\oint_\Omega \nabla^2 p\ \Omega = \oint_\Omega \nabla \cdot \boldsymbol{u}^p\ \Omega \quad \xrightarrow{\text{Gauss}} \quad \oint_S \nabla p\ \boldsymbol{n} dS = \oint_S \boldsymbol{u}^p\ \boldsymbol{n} dS, \tag{4.50}$$

Considering the properties constant throughout each particular cell faces, equation 4.50 can be rewritten as:

$$\sum_{f=1}^{m} \nabla p_f\ \boldsymbol{n}_f S_f = \sum_{f=1}^{m} \boldsymbol{u}_f^p\ \boldsymbol{n}_f S_f, \tag{4.51}$$

The term on the right is the volumetric flow of the predictor velocity through face f, while the term on the left can be approximated according to equation 4.9:

$$\sum_{f=1}^{m} \frac{P_{NB} - P_P}{\delta_{nf}} S_f = \sum_{f=1}^{m} \dot{q}_f^P, \tag{4.52}$$

Reordering terms, we obtain an algebraic equation for pressure:

$$a_P P_P + \sum_{f=1}^{m} a_{NB} P_{NB} = b_P, \tag{4.53}$$

where the equation coefficients derived from the Laplacian term depend only on geometrical parameters while the independent term, b_P, dervied from the divergence integration, depends only on the predcitor velocity:

$$a_P = -\sum_{f=1}^{m} \frac{S_f}{\delta_{nf}}, \tag{4.54}$$

$$a_{NB} = \frac{S_f}{\delta_{nf}}, \tag{4.55}$$

$$b_P = \sum_{f=1}^{m} \dot{q}_f^P, \tag{4.56}$$

Since the non-homogeneous Poisson equation is an elliptic problem, the same type of boundary conditions than the ones derived for the predictor velocity apply (see Section 4.3.4).

4.3.6 Pressure correction

The original Fractional-Step formulation proposed by Chorin [9, 14] was numerically and analytically second-order convergence in velocity, whereas the pressure was only first-order accuracy in time [15, 16]. When evaluating the predictor velocity through explicit time integration schemes, this problem is not a major drawback given that the CFL constraint imposes a small time step for stability reasons. This small time step minimizes the convergence error even for a first-order accuracy method. However, implicit methods usually use larger time steps, and the first-order accuracy for pressure may not be acceptable. It has been speculated in the literature whether this characteristic was inherent to the method [17, 18] or not. However, the projection methods accuracy can be increased through a pressure correction method [19].

Pressure correction methods are based on adding an estimated pressure gradient, ∇p_*, to the predictor velocity equation (See Equation 4.23). This additional term brings the solution of $\boldsymbol{u}^P$ much closer to its solenoidal part $\boldsymbol{u}$ since all the terms of the original equation are taken into account for the $\boldsymbol{u}^P$ evaluation. Since the estimated pressure gradient term is an approximation, a final correction has to be performed to obtain the exact solenoidal solution. The mathematical procedure is hereafter detailed.

Initially, the predictor predictor velocity equation (See Equation 4.23) is temporally discretized through an implicit backward Euler scheme, and the estimated pressure gradient at $n+1$, ∇p_*^{n+1}, added to the convective and diffusive terms (Equation 4.58). If we subtract this equation to the original time-discretized RANS (Equation 4.57), we obtain Equation 4.59:

$$\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^{n}}{\Delta t} = -\left(\boldsymbol{u}^{n+1}\cdot\nabla\right)\boldsymbol{u}^{n+1} + \nabla\cdot\left[2\nu S\left(\boldsymbol{u}^{n+1}\right)\right] - \nabla p^{n+1} \tag{4.57}$$

$$-$$

$$\frac{\boldsymbol{u}^{P,n+1}-\boldsymbol{u}^{n}}{\Delta t} = -\left(\boldsymbol{u}^{n+1}\cdot\nabla\right)\boldsymbol{u}^{n+1} + \nabla\cdot\left[2\nu S\left(\boldsymbol{u}^{n+1}\right)\right] - \nabla p_{*}^{n+1} \tag{4.58}$$

$$\frac{\boldsymbol{u}^{n+1}-\boldsymbol{u}^{P,n+1}}{\Delta t} = -\nabla\delta p^{n+1} \tag{4.59}$$

where $\nabla\delta p^{n+1}$ is the gradient of the difference between the estimated pressure, p_{*}^{n+1}, and the real pressure, p^{n+1}:

$$\nabla p^{n+1} - \nabla p_{*}^{n+1} = \nabla\left(p^{n+1}-p_{*}^{n+1}\right) = \nabla\delta p^{n+1}, \tag{4.60}$$

Applying the divergence to Equation 4.59 and taking into account the incompressibility constrain, we obtain a Poisson equation for the difference between the estimated and the real pseudo-pressure:

$$\frac{1}{\Delta t}\cancelto{0}{\nabla\cdot\boldsymbol{u}^{n+1}} - \frac{1}{\Delta t}\nabla\cdot\boldsymbol{u}^{P,n+1} = -\nabla\cdot\nabla\delta p^{n+1}, \tag{4.61}$$

$$\downarrow$$

$$\nabla\cdot\boldsymbol{u}^{P,n+1} = \nabla^{2}\delta\tilde{p}^{n+1}, \tag{4.62}$$

Then, the divergence-free velocity field can be recovered through Equation 4.59, which is essentially a Helmholtz-Hodge decomposition.

$$\boldsymbol{u}^{n+1} = \boldsymbol{u}^{P,n+1} - \nabla\delta\tilde{p}^{n+1}, \tag{4.63}$$

where $\delta\tilde{p}^{n+1} = \Delta t\delta p^{n+1}$. Finally, the last remaining step to perform, is recovering the real pressure field at $n+1$ by applying Equation 4.60 and canceling the gradient operator:

$$p^{n+1} = p_{*}^{n+1} + \delta p^{n+1}, \tag{4.64}$$

In the present implementation, the pressure estimation is performed according to the expression below [20], which allows a fast convergence of the velocity field (i.e. convective coefficients) within the implicit inner iteration loop:

$$p_*^{n+1,k} = \frac{1}{2}\left(p^n + p^{n+1,k}\right), \tag{4.65}$$

The superindex k corresponds to the pressure obtained in the implicit inner loop (within the same time-step) used to linearize the convective term.

Therefore, the independent term of the algebraic equation for the predictor velocity (Equation 4.47) has to be modified accordingly to take into account the estimated pressure term:

$$b_{iP}^k = \frac{\rho\Omega}{\Delta t}u_{iP}^n - \frac{\Omega}{\Delta t}\left(\nabla p_{*iP}^{n+1,k}\right) = \frac{\rho\Omega}{\Delta t}u_{iP}^n - \frac{\Omega}{2\Delta t}\left(\nabla p_{iP}^n + \nabla p_{iP}^{n+1,k}\right), \tag{4.66}$$

4.3.7 Linear equation system solvers

In the algorithm presented in Section 4.3 for the numerical resolution of the RANS equations, there are two stages in which a linear set of equations has to be solved. The first one is for the evaluation of the predictor velocity, $\boldsymbol{u}^P$, while the second one is the discrete Poisson equation for pressure. The characteristics of each system of equation are different, so they are their resolution methodologies. To describe them, a discretized domain of N total nodes will be considered. These N nodes can be split into n inner nodes, and m boundary nodes such that $N = n + m$.

Predictor velocity equation system resolution

The discrete linear equation of the predictor velocity for a generic inner node, has been previously obtained in section 4.3. For the sake of clarity, the equation is recalled hereafter:

$$a_{iP}^k u_{iP}^{P,k+1} + \sum_{f=1}^{m} a_{iNB}^k u_{iNB}^{P,k+1} + a_{jP} u_{jP}^{P,k+1} + \sum_{f=1}^{m} a_{jNB} u_{jNB}^{P,k+1} = b_{iP}^k,$$

where a_{iP}^k, a_{iNB}^k, a_{jP}, a_{jNB} and b_{iP}^k are constant coefficients which are known at the time of solving the equation set. On the other hand, for boundary nodes, the equations will depend on the specified boundary condition type at each node, Neumann or Dirichlet.

Therefore, a predictor velocity equation system for a given component, u_i^P, will have N equations, from them, n equations for the inner grid points, and m equations for the boundary nodes. All inner nodes will have an equation of the form described

above, while at the boundaries, the equation type will depend on the specified boundary condition at each specific point.

It is important to point out that the inner node equation of a given component, u_i^P, is coupled with the other variables of the problem, u_j^P. This coupling is due to the diffusive cross terms $a_{j_P} u_{j_P}^{P,k+1}$ and $\sum_{f=1}^{m} a_{j_{NB}} u_{j_{NB}}^{P,k+1}$. There are two different ways of dealing with this issue. The first one is considering a set of equations of size $3N$ in which all the problem variables (the three velocity components) are solved at once. However, this approach has the inconvenient that the system matrix size is three times the number of mesh nodes. Therefore, a new topology for the system matrix has to be created while a pre and post processing step has to be performed at each solver execution to transfer data from the variable containers of size N to a matrix and solution vector of size $3N$.

Another method for solving the variable coupling is based on including the cross terms as a source term. If we consider the variables $u_{j_P}^{P,k+1}$ and $u_{j_{NB}}^{P,k+1}$ at the inner iteration k instead of at iteration $k+1$, the resulting terms $a_{j_P} u_{j_P}^{P,k}$ and $\sum_{f=1}^{m} a_{j_{NB}} u_{j_{NB}}^{P,k}$ can be deferred to the term $b_{i_P}^k$. Hence, the problem variables become uncoupled, and each velocity component can be solved separately. In this case, three equation systems of size N must be solved, without needing any specific pre and post process of the solution. In our case, both methodologies were tested, being the latest one adopted due to efficiency reasons.

Once the equation set and its coupling resolution method are determined, we obtain a set of sparse matrices and their corresponding independent term vectors that have to be solved through a linear solver method. Since the sparse matrix for the predictor velocity has no special property aside sparseness, the iterative Generalized Minimal Residual Method (GMRES) has been used to obtain a solution [21, 22]. Additionally, the matrix coefficients vary at each inner iterative loop due to their dependency on the convective fluxes, and therefore, it makes no sense the use of a direct solver with an expensive pre-processing of the solver matrix. The GMRES method is suitable for non-symmetric equation systems, and it is especially efficient for sparse matrices, being a suitable option for the present case.

Pressure equation system resolution

As determined in Section 4.3.1, the equation for pressure is a Poisson equation. Since it features a symmetric Laplacian operator, if a symmetry preserving discretization method is used, the discrete operator conserves this characteristic property. As described in Section 4.3.5, by applying a central difference scheme for the gradient operator, we obtain a the following discretized equation for the inner nodes:

$$a_P P_P + \sum_{f=1}^{m} a_{NB} P_{NB} = b_P,$$

The coefficient a_{NB}, which defines the relation between a given node and its neighbors, it only depends on geometrical terms $a_{NB} = S_f/\delta_{nf}$. For a given face, the term is identical for the two adjacent nodes, which ensures that the discrete matrix for the inner nodes is symmetric (see Section 4.2).

However, the information of the boundary nodes must be still introduced. If the boundary node equations were explicitly included in the system, the symmetry of the matrix would be broken. To preserve the symmetry property, the boundary conditions will be introduced through the inner node equations whose associated cell owns at least one boundary face. According to Equation 4.52, the discrete pressure gradient at any cell face can be expressed as in equation below. For a boundary face, P_{NB} is the pressure at the face node. If we consider a Neumann condition ($\partial P/\partial n = 0$) for this face, the condition would be expressed as $P_{NB} = P_P$:

$$\left.\frac{\partial P}{\partial n}\right|_{n=0} \approx \frac{P_{NB} - P_P}{\delta_{nf}} \quad \xrightarrow[P_{NB}=P_P]{\partial P/\partial n=0} \quad \frac{P_{NB} - P_{NB}}{\delta_{nf}} = 0,$$

Therefore, by suppressing the contribution of the boundary face to the inner node equation, the Neumann condition would be already taken into account.

This methodology allows conserving the symmetry properties of the system matrix whose size would be n (the number of inner nodes). This approach has the inconvenient that data containers are of size N (they include the boundary nodes), and a pre and post processing step is required to transfer information from the variable vectors to the system and vice versa. Nonetheless, the costs of these processes are largely offset by the great advantage that offers a symmetric matrix regarding the linear solver efficiency. The symmetry of the system matrix allows using the Conjugate Gradient linear solver algorithm which also requires the matrix to be positive-definite[23]. This method is specially efficient for small condition number matrices. Although the Conjugate Gradient method is iterative, a direct solver could also be used given that the matrix only depends on geometric parameters and it remains constant for fixed geometries and meshes.

4.4 Resolution algorithm

In this section, the resolution algorithm of the numerical process described in previous sections is presented and summarized in the flow chart in Figure 4.4. In the following table, a summary of notation is given to easily understand the diagram:

Table 4.1: Summary of notation for flow chart 4.4.

Symbol	Definition
$()^n$	quantity at temporal position n
$()^k$	quantity at implicit inner iteration position k
$()^P$	predictor
$()_P$	evaluated at cell node
$()_{NB}$	evaluated at neighbor cell node
$()_f$	evaluated at cell face
$()_i$, $()_j$	vectorial subindex notation.
m	number of faces of a given cell
ε	required precision

Since the time integration scheme is implicit, it entails that a variable evaluated at any inner iteration $k+1$, it is also evaluated at the temporal step $n+1$. Therefore, for the sake of clarity, the superindex $n+1$ will be suppressed in the algorithm for all terms evaluated at $k+1$. By contrast, it has to be pointed out that variables at $n+1$ might be also evaluated at k except for the first iteration.

On the other hand, the yellow frame groups all the steps of the inner implicit loop. This inner cycle (within a given time step) is performed in order to linearize the predictor velocity algebraic equation, as explained in Section 4.3.4. It is interesting to point out that in Equation 4.42, the equation coefficients, a, are evaluated with magnitudes obtained in the inner iteration k, while the equation variables, u_i^P, are defined at $k+1$. Finally, once the velocity is converged within the time step $n+1$ (decision box in the flow chart), the system is advanced in time, from $n+1$ to $n+2$ through the outer loop.

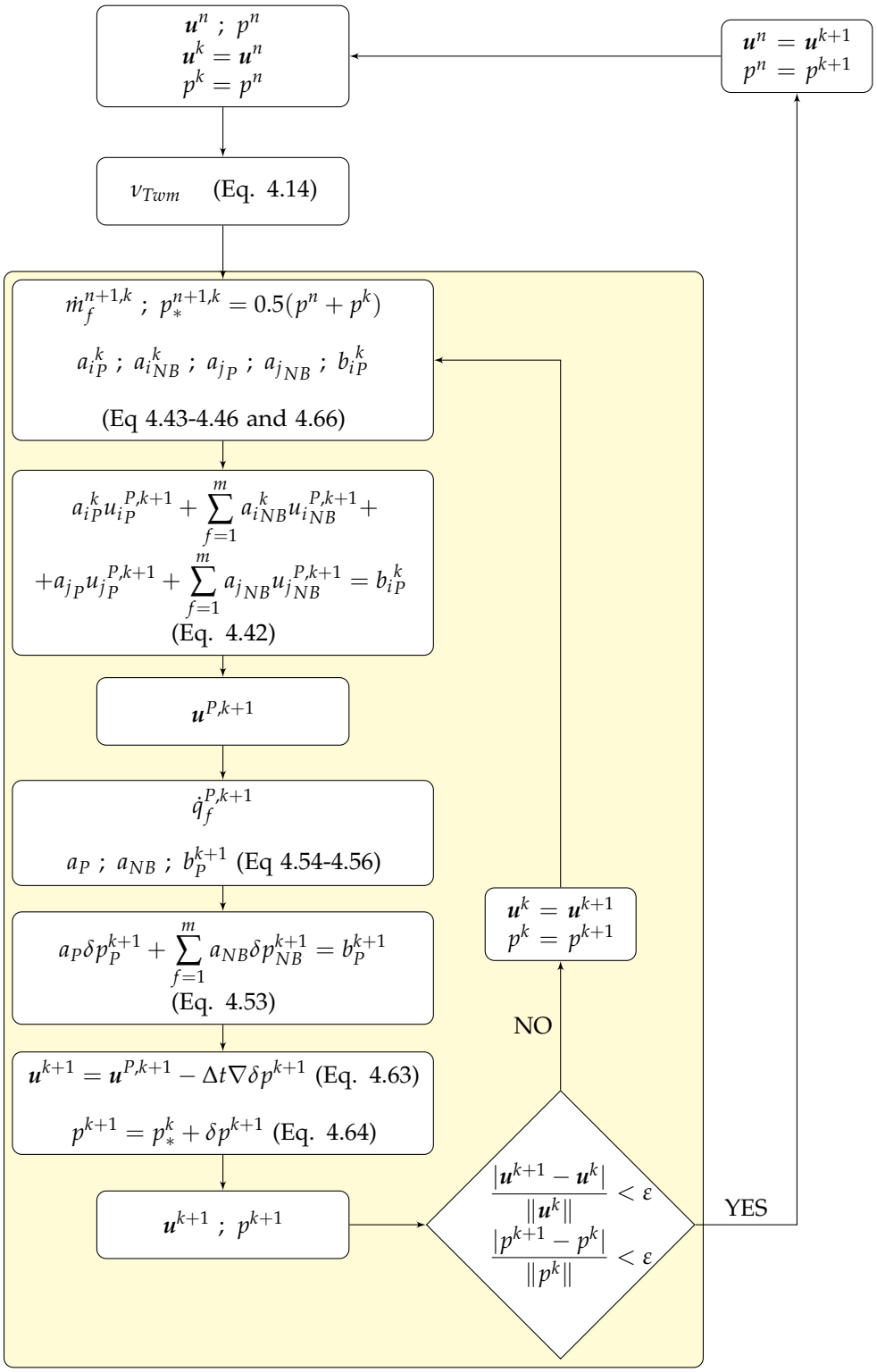

Figure 4.4: Flow chart of the implicit numerical resolution of RANS equations.

4.5 Conclusions

In this Chapter, the mathematical and numerical strategy used for the present TLM has been developed. The RANS equations have been discretized through the Finite Volume Method in order to obtain an algebraic set of equations allowing to obtain a numerical solution for the elliptic PDE. The concepts of continuous operator property conservation in the discrete domain and its consequences in the physical model have been briefly explained.

Additionally, a fractional-step method has been used in order to resolve the velocity-pressure coupling and to enforce the incompressibility constraint. Projection methods are only one-order accuracy for the pressure. In general, this accuracy degree is sufficient for small time-steps. However, since one of the purposes of the TLM is allowing the use of large time-steps, an improvement of the method accuracy is needed. For that reason, a pressure-correction is applied to the fractional-step basic algorithm. On the other hand, the use of large time steps forces the use of implicit schemes for the temporal integration of the TLM governing equations due to stability reasons.

Finally, at the end of the Chapter, the global numerical algorithm including all the resolution steps is summarized with a flow chart.

Bibliography

[1] R. W. C. P. Verstappen and A. E. P. Veldman. Symmetry-preserving discretization of turbulent flow. *J. Comput. Phys.*, 187:343–368, 2003.

[2] F. X. Trias, O. Lehmkuhl, A. Oliva, C. D. Pérez-Segarra, and R. Verstappen. Symmetry-preserving discretization of Navier-Stokes equations on collocated unstructured grids. *J. Comput. Phys.*, 258:246–267, 2014.

[3] B. Sanderse. Energy-conserving Runge-Kutta methods for the incompressible Navier-Stokes equations. *J. Comput. Phys.*, 233:100–131, 2013.

[4] F. Capuano, G. Coppola, L. Randez, and L. De Luca. Explicit Runge-Kutta schemes for incompressible flow with improved energy-conservation properties. *J. Comput. Phys.*, 328:86–94, 2016.

[5] R. LeVeque. *Finite Volume Methods for Hyperbolic Problems.* Cambridge Univesity Press, 2002.

[6] E.F. Toro. *Riemann Solvers and Numerical Methods for Fluid Dynamics.* Springer-Verlag, 1999.

[7] M. Hazewinkel. *Positive-definite form.* Encyclopaedia of Mathematics, Springer, 2001.

[8] I. Jacques and C. Judd. *Numerical Analysis.* Chapman and Hall, New York, 1987.

[9] A.J. Chorin. Numerical solution of the Navier-Stokes equations. *J. Comput. Phys.*, 22:745–762, 1968.

[10] S. V. Patankar. *Numerical Heat Transfer and Fluid Flow.* Hemisphere Publishing Corporation, 1980.

[11] F. Moukalled and M. Darwish. A comparative assessment of the performance of mass conservative-based algorithms for incompressible mulitphase flows. *Numer. Heat Tr. B-Fund.*, 42:259–283, 2002.

[12] L. Koenigsberger. *Hermann von Helmholtz*. Clarendon Press, 1906.

[13] J. Bladel. *On Helmholtz's Theorem in Finite Regions*. Midwestern Universities Research Association, 1958.

[14] A.J. Chorin. On the convergence of discrete approximations to the Navier-Stokes equations. *Math. Comput.*, 23:341–353, 1969.

[15] W. E and J. Liu Guo. Finite difference schemes for incompressible flows in the velocity-impulse density formulation. *J. Comput. Phys.*, 130:67–76, 1997.

[16] R. Temam. Remark on the pressure boundary condition for the projection method. *Theoret. Comput. Fluid Dynam.*, 3:181–184, 1991.

[17] J.B. Perot. An analysis of the fractional step method. *J. Comput. Phys.*, 108:51–58, 1993.

[18] J.C. Strikwerda and Y.S. Lee. The accuracy of the fractional step method. *SIAM J. Numer. Anal.*, 37:37–47, 1999.

[19] D.L. Brown, R. Cortez, and M.L. Minion. The accuracy of the fractional step method. *J. Comput. Phys.*, 168:464–499, 2001.

[20] A. Carmona, O. Lehmkuhl, C. D. Pérez-Segarra, and A. Oliva. Numerical analysis of the transpose diffusive term for viscoplastic-type non-newtonian fluid flows using a collocated variable arrangement. *Numer. Heat Tr. B-Fund.*, 67:410–436, 2015.

[21] Y. Saad and M.H. Schultz. Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM J. Sci. Stat. Comput.*, 7:856–869, 1986.

[22] R. Borrell Pol. Parallel algorithms for computational fluid dynamics on ustructured meshes. *Doctoral thesis. Universitat Politècnica de Catalunya*, 2012.

[23] M.R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stand.*, 49(6):409–436, 1952.

5

Two-Layer wall model implementation

In this Chapter, the implementation allowing to solve the numerical model presented in Chapter 4, is detailed. The TLM is a complete CFD solver which resolves the near-wall flow according to its own physical and mathematical model, in this case, the Unsteady Reynolds-Averaged Navier-Stokes (URANS) equations. The TLM solver interacts bidirectionally with the main CFD solver, which solves the whole computational domain through an LES approach. Therefore, an algorithmic infrastructure allowing not only to solve the mathematical and numerical model but also dealing with the TLM/LES solver coupling, had to be developed.

The new algorithm is implemented in the existing C++-based unstructured parallel code TermoFluids[1] as a new module, allowing WMLES computations for non-equilibrium flows which were not previously available. Since TermoFluids is a parallel code, an efficient parallelization strategy has also been developed for a correct management of computational resources. At the end of the present Chapter, the performance of the parallel algorithm is evaluated through a standard strong scalability test.

5.1 Preliminary concepts

In this section, several specific terms related to parallel algorithms will be used. Before developing the section contents, a brief description of the standard domain decomposition strategy and the main related concepts will be provided to facilitate the reading of the document.

The standard domain decomposition is a methodology for the parallel resolution of partial differential equations (PDEs). It is based on dividing the initial discretized domain into n subdomains with a similar number of discrete elements. Then, these

subdomains or partitions are subsequently distributed among n parallel processes, each one owned by an independent physical core. The parallel distribution concept refers to the whole computational domain and the way it has been partitioned. The subset of discrete elements assigned to a given parallel process is referred to as its owned elements, while the rest are labeled as external. Thus, for each parallel process, we may talk about owned cells, owned nodes, owned components of a data container, external nodes, external faces, et cetera. Since the discrete PDE variables at each subdomain are related to the values of its neighboring processes, to perform parallel computations, the transmission of data between the parallel subdomains becomes necessary. Halo elements are the external elements required by any parallel process in order to solve the PDE within its assigned domain. In general, halo elements are obtained from neighboring subdomains using network communications. These communications are called "halo updates".

The parallelization concepts described above will be hereafter applied to the algorithm presented in the previous chapter for the numerical resolution of the URANS PDE.

For the TLM solver, a software platform has been developed to efficiently perform the algebraic operations required by the finite volume method (FVM). The TermoFluids original code, which is the base for the present implementation, is a fully operational CFD code optimized for Direct Numerical Simulations and Large Eddy Simulations of geometrically complex flows. It features several mathematical and numerical tools that have been used during the implementation of the present code, such as topologies for parallel communications or an extensive library of linear equation system solvers [1, 2].

In Figures 5.1 and 5.2, the main concept of TLM is depicted. The wall model has to be implemented in an environment based on a fluid domain discretized with an unstructured mesh (LES mesh) and bounded by at least one solid face on which a boundary layer may be developed. In the example displayed in Figure 5.1, the flow around an Ahmed car body is shown. The solid boundary is a simple car model, which generates a fully 3D boundary layer which remains attached to the body surface until the car back. At this point, the flow separates from the car surface, causing a strong flow recirculation. Figure 5.2 shows the same simulation but featuring the TLM mesh, which wraps the solid boundary as a shell. In Figure 5.3, a detail of the TLM mesh and its surroundings is displayed. It can be observed that the TLM mesh is generated by extruding the solid boundary surface grid in the wall-normal direction towards the LES domain. Then, the URANS equations are solved within this mesh in order to obtain an accurate near-wall velocity field, which is also shown in the figure. In Figure 5.4, the same concept is depicted for a DU 91-W2-250 airfoil geometry. In this case, the LES mesh is displayed for wall resolved (left) and wall modeled (right) LES simulations to allow comparison of the required grid densities for each modeling

strategy. In this case, only a 2D section of the LES mesh is displayed for the sake of clarity, while the TLM grid is fully shown.

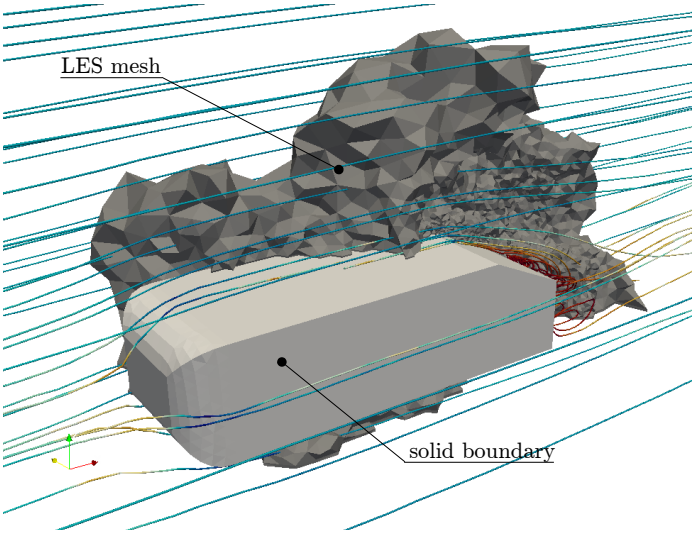

Figure 5.1: Flow around an Ahmed car body. A sample of the LES grid as well as the body solid surface are displayed in the figure together with the flow streamlines.

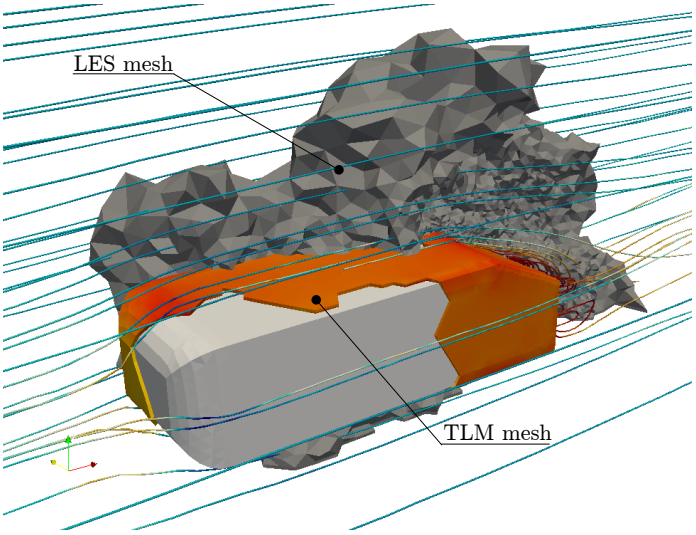

Figure 5.2: Flow around an Ahmed car body. A sample of the LES mesh is shown together with the TLM grid wrapping the Ahmed body surface.

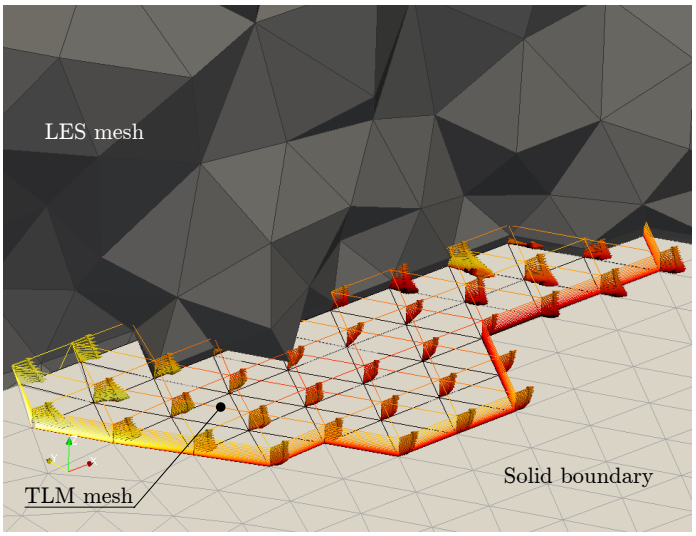

Figure 5.3: Detail of the LES and TLM grids, including the near-wall velocity vector field of a flow around an Ahmed car body.

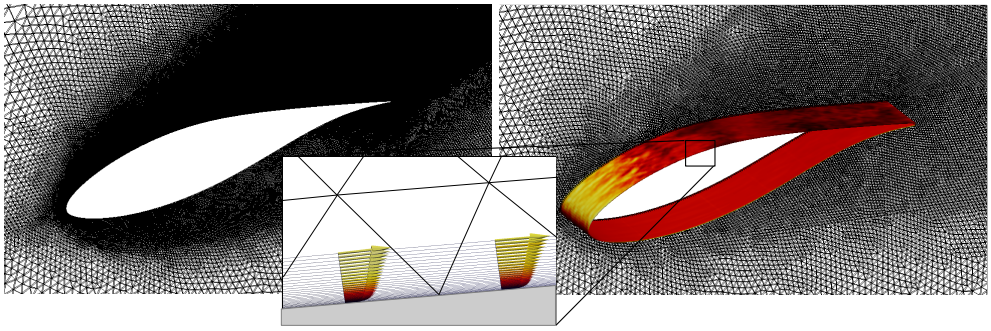

Figure 5.4: Flow around a DU 91-W2-250 airfoil. Snapshots of the wall resolved LES mesh (left) and the WMLES mesh with the embedded TLM grid, colored by velocity magnitude (right). A detail of the WM near-wall grid and the velocity vector field is also displayed (thumbnail).

5.2 TLM mesh generation and partitioning

An essential element for the numerical resolution of the URANS equations is the computational grid, which represents the discretized spatial domain in which the equations will be solved. Additionally, it has to provide all the features required by the finite volume method, such as extensive geometrical data, global and local element identifiers for faces, vertices, cells and nodes, grid element iterators, parallel communication tools etc.

Initially, the geometrical grid generation and its partition will be explained. The starting point is a solid boundary of the LES domain, which is superficially discretized with an unstructured pattern, and distributed through different processors according to the LES domain partition. In Figure 5.5, a sketch of the initial solid surface is displayed.

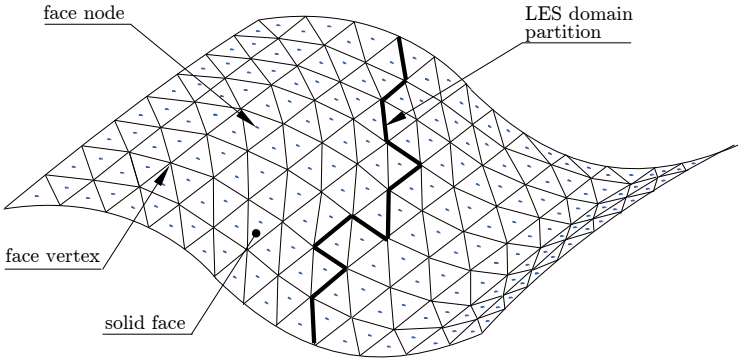

Figure 5.5: Discretized LES domain solid boundary. The domain partition is represented by a thick continuous line.

The TLM mesh has to be constructed from this initial surface by extruding it up to a given height. The extrusion length is an important problem parameter, and it can be either the first off-wall LES node height, or a custom height that can be freely chosen by the user.

5.2.1 Parallelization strategy

Regarding the parallel distribution, the most straightforward implementation would be using the same as the LES domain for the TLM grid. This would make the algorithm significantly simpler than considering a specific distribution for the TLM mesh. In this case, only the parallel processes owning a part of the solid face would be involved in the mesh generation, and therefore, in the numerical resolution of the URANS equations. The communications between the LES and the TLM domains would be almost straightforward, since the LES nodes and their TLM counterparts would be owned by the same processors. This approach was initially chosen. However, it proved to be extremely inefficient in most cases, especially for those with small solid boundaries compared to the whole computational domain size. In some cases, less than 50% of the total number of processors in the LES partition owned a solid boundary part, and thus, worked in solving the wall model algorithm. This caused an extreme work-load imbalance among the parallel processes. On the other hand, the communication cost between the LES and the TLM domains was relatively low since these communications were only performed through the TLM grid boundary nodes, which represents a small amount of parallel links compared to a three-dimensional

partition. In the trivial partition strategy, communications between the LES and the TLM domain were only required in case the mesh was extruded beyond the first off-wall nodes. Otherwise, the top and the bottom TLM boundaries would be owned by the same LES partitions such that the data transfer would be straightforward.

Hence, different parallel distributions were applied to each particular domain, the LES and the WM one. A specific parallel distribution using the total amount of available cores was used for the TLM domain. Additionally, a communication strategy was developed for the information flow between the LES and the TLM computations.

In the sections below, more details about the communication strategy between these two domains will be given. A scalability performance test comparing the two different TLM parallel distributions will be carried out in Section 5.4 in order to evaluate the gains obtained by using and specific partition for the TLM mesh.

5.2.2 TLM mesh partitioning

Initially, the solid surface geometrical and topological information is distributed across the LES parallel distribution. Therefore, this information has to be collected and redistributed to a new parallel distribution scheme, in which the work-load associated with the TLM domain has to be properly balanced across all the available processors. In Figure 5.6, this redistribution is sketched.

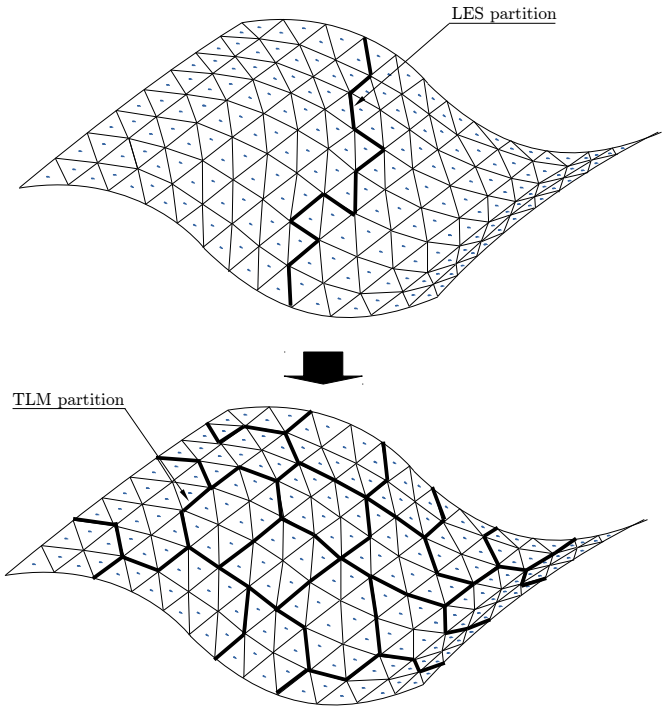

Figure 5.6: Solid face mesh partition of the LES (top) and TLM (bottom) domains. Both partitions feature the same number of processors, but in general, in the LES partition, most of processes are away from the wall and do not own any solid boundary part.

To carry out the new balanced work-load distribution, the external Metis code has been used [3]. The Metis code features a graph partitioning algorithm which pursues a balanced number of graph nodes per process while minimizing the number of graph edges broken by the partition.

Before proceeding with the code development process, the topology concept and its related terms will be explained. A computational mesh can be topologically described as a graph, in which each cell is a node of the graph. Each node of the graph will be connected to at least one neighboring node, corresponding to the neighboring cells contiguous to its cell faces. In order to construct the graph associated with a computational grid, each node is labeled with a global ID, a number that is unique for the whole computational domain. The graph edges establish which nodes are

connected to each other. In Figure 5.7 the generation of a graph from a schematic 2D mesh is depicted.

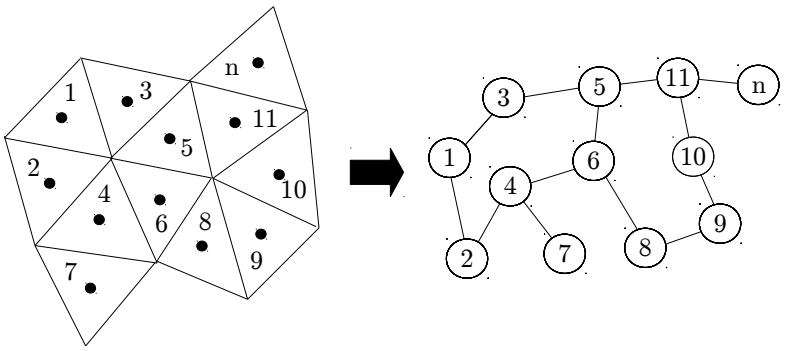

Figure 5.7: Left: computational mesh with its corresponding nodes and cells, labeled with a global ID. Right: Graph associated with the mesh on the left.

Once the graph of the solid boundary mesh has been generated, it is partitioned by the Metis code. As commented above, the main targets of the partition process are a balanced workload distribution among processors, with a minimum partition of graph edges. While the first objective is the main one and may seem trivial, the second one is also important. Each broken edge means a new communication among processors that has to be repeatedly performed along the computation, and a large number of them may entail a significant loss of efficiency. In Figure 5.8, the graph obtained in Figure 5.7 is partitioned in two different ways. In both examples, the graph nodes are uniformly distributed between two different groups, each one containing six elements. However, the example on the left is optimally distributed since only three edges are broken, which is the minimum necessary amount of parallel links to perform an optimal node repartition. On the other hand, in the example on the right, five edges are broken to achieve the same workload balance.

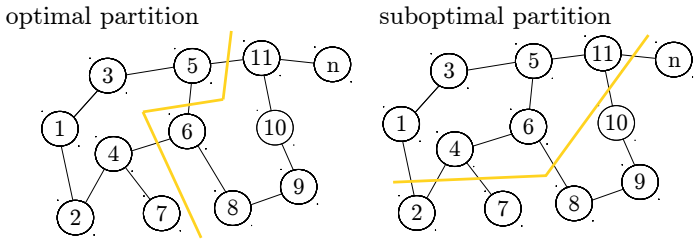

Figure 5.8: Graph partitions. The yellow line represents the division between the two processes. Left: optimal solution with a balanced workload and 3 partitioned edges. Right: suboptimal solution with a balanced workload and 5 partitioned edges.

The input data for the Metis algorithm is a set of stencils. A stencil is a set of global IDs containing information about a given node and its connections with the rest. The first global ID is the identifier of a specific node, while the rest of IDs correspond to the neighboring nodes at which the main one is connected to. Therefore, a stencil has to be provided for each graph node, which allows the Metis code to reconstruct the full graph before carrying out the partition process.

Since the initial data is distributed along the LES partition, all processes owning part of the solid boundary are responsible for collecting all the stencils of its surface part. The stencils corresponding to nodes located at an existing mesh partition limit will have to include the halo nodes in order to keep correct grid connectivity. Once this task is finished, all the stencils are conveniently packed and sent through a "Message Passing Interface" (MPI)[4] communication to the master core (core 0). Once the processor 0 receives all the stencil information from the whole computational domain, it executes sequentially the Metis algorithm, which produces an output file that can be read by all the processes involved in the MPI execution.

With the information provided by the Metis output file, a topology can be created. A topology is set of node global IDs, and contains information about their distribution along the parallel processes. Additionally it also contains data about halo nodes, i.e., not-owned nodes connected to the owned ones among which information exchange can occur. This information is sufficient to establish the parallel communications necessary to guarantee the connectivity of the whole domain. On the other hand, the topology also contains local IDs, which are node identifiers only valid within each particular partition. While the global IDs can be any positive integer number, the local IDs range from 0, to $k-1$, being k the total number of nodes of a particular partition including the halo ones. Taking the graph distribution example in Figure 5.8 on the left, its corresponding topology would be as follows:

Table 5.1: Global and local ID distribution between processors corresponding to the topology displayed in Figure 5.8 on the left.

Partition 1		Partition 2	
owned nodes			
global ID	local ID	global ID	local ID
1	0	6	0
2	1	8	1
3	2	9	2
4	3	10	3
5	4	11	4
7	5	n	5
halo nodes			
global ID	local ID	global ID	local ID
6	6	5	6
11	7	4	7

Through the halo node global IDs, each process can identify at which other processes it is connected to by searching the partition that owns its halo global IDs. In this trivial example, Process 1 has to establish a link for its halo nodes 6 and 11 with Process 2, who owns these global IDs.

At this point, all process know which superficial nodes own and how they are related to each other through the stencil information. This allows the redistribution of the minimum necessary amount of geometrical information needed to generate the TLM mesh. This geometrical data concerns superficial node and vertex positions, surface normals at vertex locations, and the vertices stencil associated to each node. The surface normal at the vertices is evaluated by averaging the normal vectors of the vertex contiguous faces. The data is efficiently packed and sent by the master process to the new TLM distribution.

5.2.3 Generating TLM mesh elements

Once the superficial geometrical and topological data has been distributed among all the processes, the 3D mesh has to be generated by extruding the initial 2D grid in the surface-normal direction. The resulting mesh will depend on three parameters: the mesh extrusion height (H), the number of extrusion layers (L), and the refinement factor (k) which controls the concentration of extruded nodes towards the wall.

The distribution of nodes along the wall-normal direction is determined according to the following hyperbolic tangent function:

$$h(l) = 2H - H\left[1 + \left[\frac{1}{\tanh(k)}\ \tanh\left[\frac{k(2L-l)}{L} - k\right]\right]\right], \tag{5.1}$$

where $h(l)$ is the height of a given layer l. The higher the value of k, the larger the concentration of the extruded nodes towards the wall.

Once the height of each layer is computed, the surface vertices can be extruded along the surface normals, obtaining a cloud of points that will be the used as a frame to create the remaining mesh elements. In Figure 5.9, the generation method of the new grid faces, cells and nodes is sketched. Additionally, a unique global ID has to be created for each different element in order to generate the topologies. Each parallel process is responsible for generating the extruded mesh corresponding to its owned solid surface part.

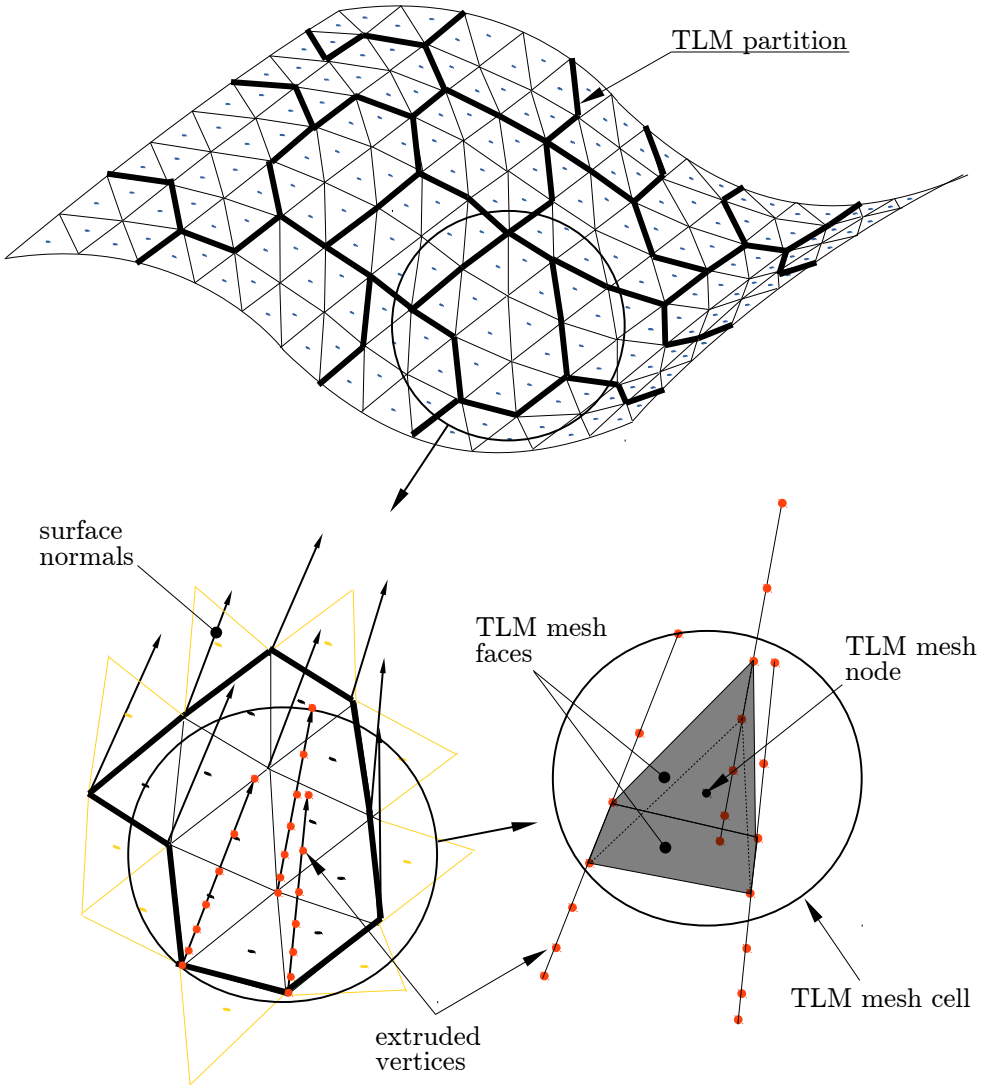

Figure 5.9: Top: TLM surface mesh distribution. Bottom left: Thumbnail of a specific partition featuring the vertex centered normal vectors and the extruded mesh vertices along them. Bottom right: A full grid cell generated from the extruded vertices, including its faces and node.

In the first step, the mesh faces are created by grouping the previously generated vertices into sets of at least three of them. Through the vertices positions, the face centroid, normal vector, and area can be computed and stored. In the second stage, the mesh cells are generated by creating a set of faces. Each cell will be composed of two wall-parallel faces, and at least three wall-normal faces, forming a three or more-sided prism, depending on the solid boundary discretization pattern. Again, once the cell vertices are determined, the cell centroid and volume can be calculated.

Finally, the mesh nodes are created. Geometrically, their positions are already defined. They can be either a cell centroid for the inner mesh nodes, or a face centroid for the boundary nodes. In this case, the global ID generation process has to be carefully addressed, since the communications among the parallel processes are carried out through the mesh nodes. Each process will enumerate its owned nodes, but it has to be ensured that the generated ID's are unique for the whole computational domain. Additionally, the halo node ID's have to be also generated and have to match the global IDs generated by their owner process. Once the enumeration of the mesh elements is finished, the communication pattern among the processes can be established by creating the corresponding topologies.

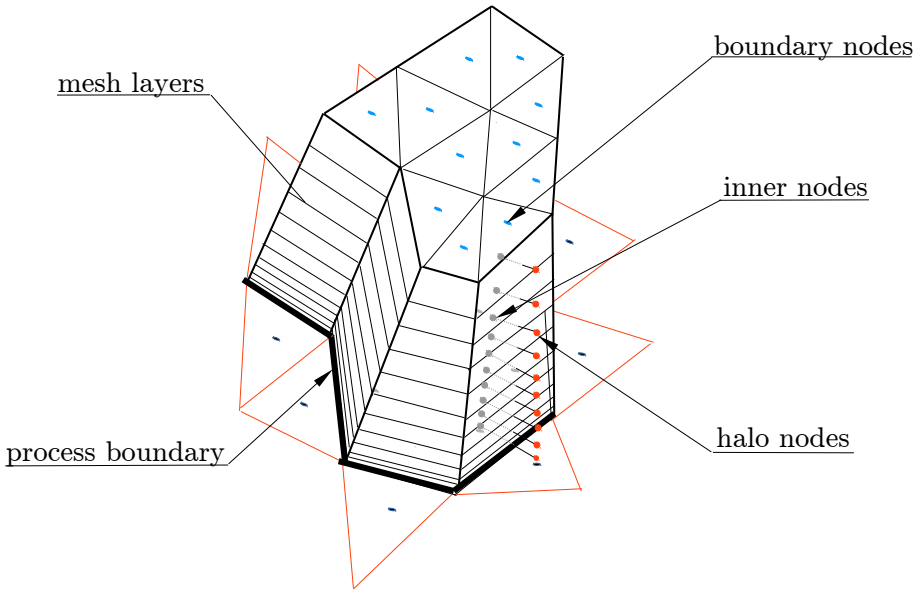

Figure 5.10: Local TLM mesh corresponding to a specific process.

The generation of the mesh elements also includes the pre-computation of all geometric magnitudes which are needed for evaluating all the equation coefficients determined in Section 4.3. This includes all the point positions (nodes, face centroids, and vertices), cell volumes, face areas, face normal vectors et cetera. At the same time, some mechanisms have to be put in place in order to perform operations, like evaluating the mass flow for a given cell etc.

5.2.4 TLM mesh boundary conditions

In order to apply different boundary conditions, the outer nodes have to be labeled in order to identify which condition has to be applied for each variable. Since the code automatically generates the mesh, a methodology had to be developed to apply the identification label (b_{ID}) at each boundary node. The top and bottom surfaces always feature the same boundary conditions (See Chapter 3), Dirichlet for the former, taking values from the LES domain, and solid wall for the latter (i.e., Dirichlet for velocities, $\boldsymbol{u} = \mathbf{0}$, and Neumann for pressure). Therefore, reserved identifiers are automatically set for these surfaces when generating the mesh.

When the TLM is applied to a closed surface such as a sphere or the Ahmed car geometry shown in Figure 5.2, the only external surfaces of the TLM mesh are the top and bottom boundaries. Therefore, there is no need for applying additional b_{ID} since all the boundary nodes are already labeled. By contrast, when the model solid surface has open edges like in the example in Figure 5.11, side boundaries are generated due to the extrusion of the surface edges. These nodes are not identified by default since the boundary condition depends on each specific case and must be labeled externally. Thus, the label of the side boundaries has to be chosen by the user.

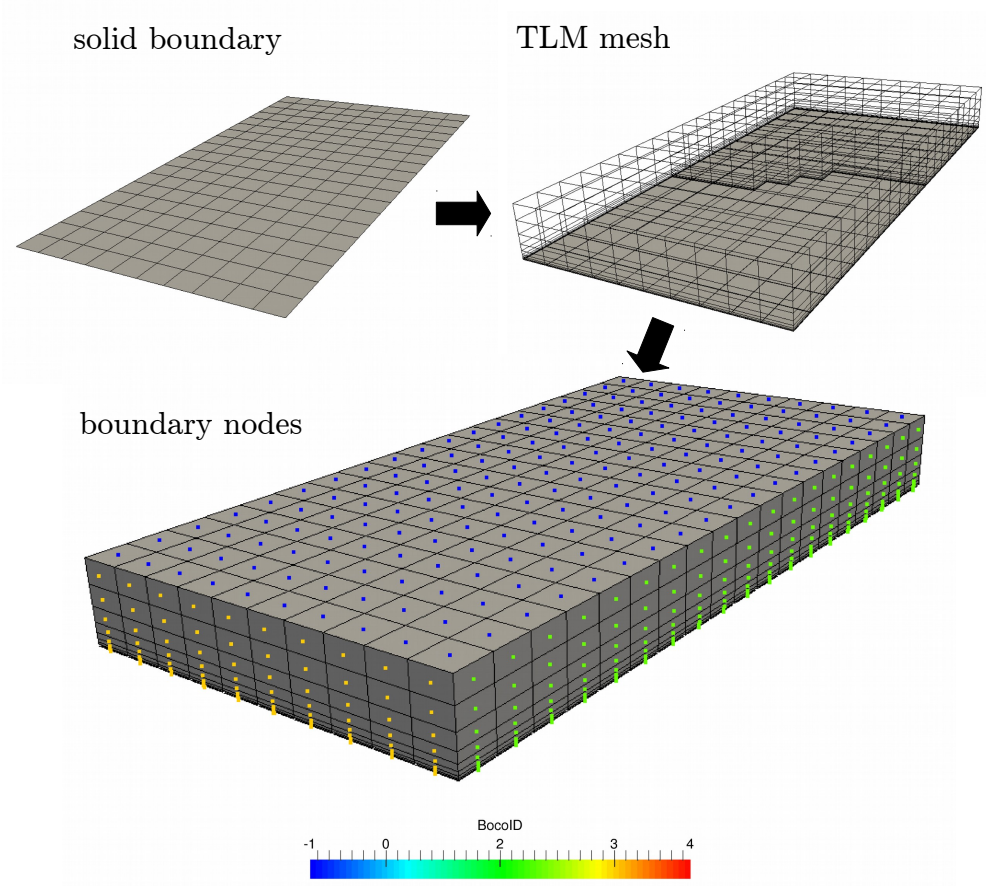

Figure 5.11: Top left: Initial discretized solid surface. Top right: Extruded mesh. Bottom: Cloud of mesh boundary nodes colored by their boundary condition identifier.

A box system has been developed to group the boundary nodes with a common b_{ID}. To define the domain containing a specific group of nodes, the coordinates of two opposite corners of the box have to be set. To facilitate the process, the b_{ID}s are applied sequentially with an order defined by the user. That means that if the bounding boxes overlap, the nodes contained by two or more boxes receive the label of the last box according to the predefined order. At the end of the process, the top and bottom boundaries are labeled with their reserved b_{ID}s, so they can be also contained

by the bounding boxes. The box procedure is illustrated in figure 5.12, where two bounding boxes are defined. As can be observed, part of the green nodes on the right are contained in box 1 and 2. Nonetheless, the final label will be the one of box 2. At the end of the b_{ID}s assignment process, a visualization file can be printed in which all the boundary nodes colored by their b_{ID} value are displayed. This helps in the setting up process, and it is also a checking tool in order to ensure that no errors have been done during the assignment process.

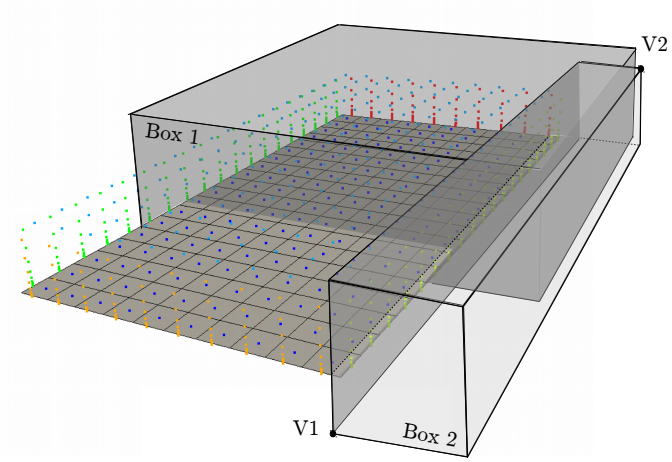

Figure 5.12: Bounding boxes intended to group a subset of boundary nodes featuring the same b_{ID}.

Once the identifier labels have been set, the particular boundary conditions are provided to the model by specifying the condition type for each variable and b_{ID}. For the present TLM, the most common boundary conditions have been implemented inflow, outflow, solid wall and periodic conditions. In the latter case, the algorithm automatically detects the existing pairs of periodic boundaries (they must be parallel), and it creates the links between the periodic nodes and their counterparts. Additionally, when needed, it creates additional halo nodes for those periodic faces which belong to different processes, while geometrical parameters such as node distances are also re-evaluated.

5.3 TLM/LES interaction

The TLM presented up to this point could work as an independent parallel CFD solver in a computational domain generated by the extrusion of a given superficial unstructured mesh. However, the purpose of the present solver is to be used as a wall model, and therefore, it has to interact with an external LES solver which has an entirely different domain and parallel distribution.

5.3.1 Domain connectivity

The interaction between the TLM and the LES solvers is bidirectional. The LES provides boundary condition to the TLM top boundary face, while the model feeds back the LES computation with a wall shear stress through the solid boundary surface.

In figure 5.13, the interaction between LES and TLM domains is sketched. The central figure shows a vertical section of the TLM and LES meshes, in which both domains are depicted in 2D. The partitions of both grids are also shown (thick continuous and dashed lines for TLM and LES domains, respectively) to understand better the parallel strategy that will be developed below. It has to be taken into account that, in general, the LES domain may extend far away from the solid boundary region, and only a few processes may be involved in the LES/TLM communication process. By contrast, for the TLM domain, all the available cores will take part in it.

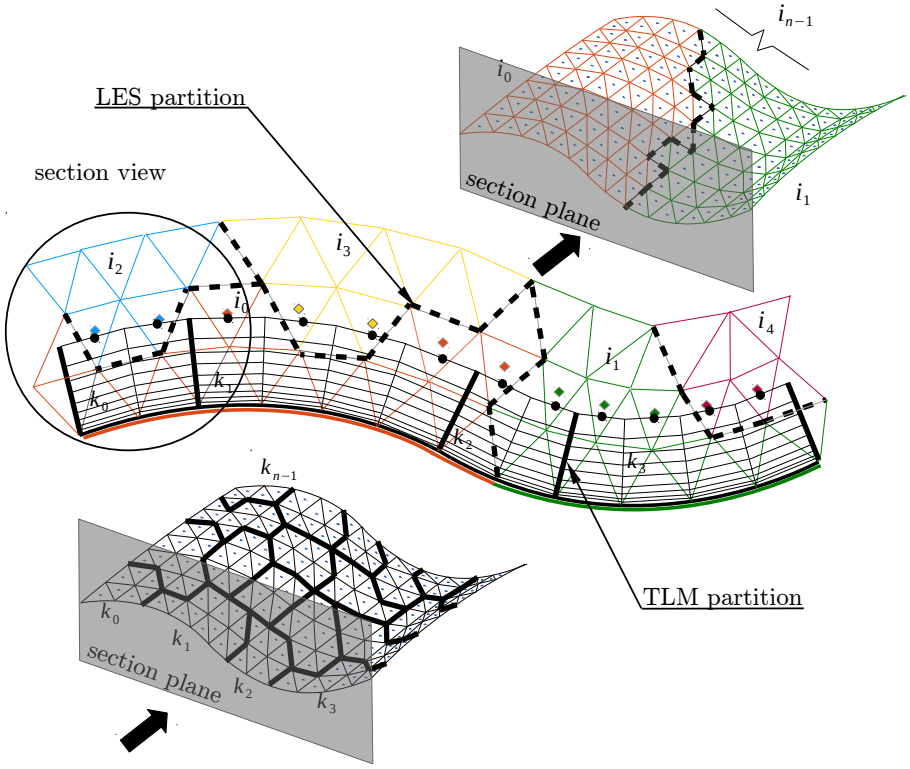

Figure 5.13: Cross-section of the TLM and LES meshes. The unstructured LES mesh is colored according to the process ownership (i). The TLM mesh partitions (k) are in black. Thick continuous lines represent the TLM parallel distribution, while the dashed thick line represents the LES parallel distribution.

The information between the LES and the TLM domains flows through two different surfaces. In Figure 5.14 a zoomed view of the mesh section is displayed. In this diagram, it can be identified the two interface regions between both computational domains. The first one is the top boundary of the TLM mesh. This surface is immersed in the LES fluid domain, and its nodes are contained in different LES cells. The information flow in this case is from the LES cell nodes to the TLM top boundary nodes. Since the method is for unstructured grids, in general, the TLM and the LES nodes are not coincident in this interface region. Therefore, a first-order interpolation

scheme is applied to determine the value of a given function, ϕ, at the TLM nodes from the information located at the LES points:

$$\phi_{TLM} = \phi_{LES} + \nabla\phi \cdot \mathbf{r}, \tag{5.2}$$

where $\mathbf{r}$ is the position vector between the LES and its corresponding TLM node, and ϕ_{TLM} and ϕ_{LES}, are the values of ϕ at the TLM and LES nodes, respectively.

It is worth noting that, it has to be identified the LES cell in which each TLM top boundary node is located. This process is not straightforward since in general, the TLM node and the LES cell do not belong to the same parallel process. The linkage between these two elements is performed during the TLM grid generation through a parallel particle method [5]. This method allows a given core to inject a particle out of its spatial domain and to get back information about the cell in which the injected particle is located, including core ownership, cell node position, and global identifiers.

Regarding the second information exchange, it occurs between the bottom TLM boundary surface nodes and the LES solid boundary points. In this case, the nodes, although distributed in different processes, they are geometrically coincident, and no interpolation process is required.

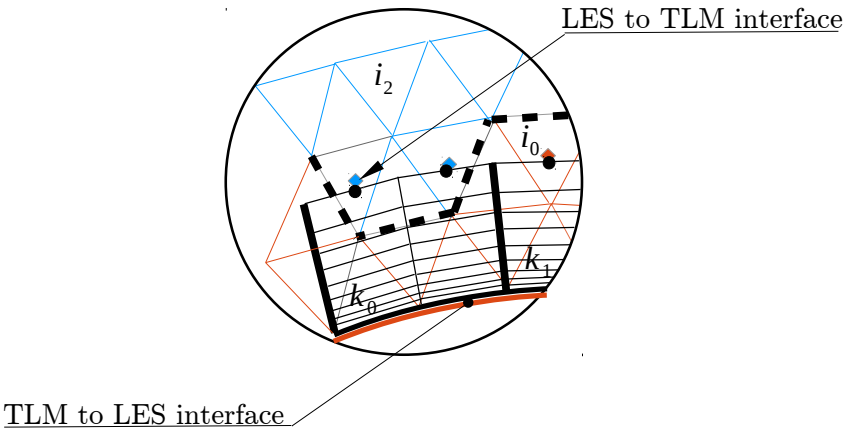

Figure 5.14: Zoomed view of Figure 5.13. The LES and the top TLM boundary nodes are displayed.

The communication scheme is the same for the two interfaces. In both cases, the interface surface has two sides (the TLM and LES one) with the same total number of points. Each point of the LES side is linked to its corresponding node at the TLM face. Nonetheless, since the parallel distribution is different for each side, each particular

process will own different parts of the interface for the LES and the TLM partitions, respectively.

The parallel problem can be reduced to a discretized surface, in which each side is partitioned with a different number of cores and with different distribution patterns. Each discrete element of a given side has its corresponding element on the other side, both labeled with the same single global ID, but in general, owned by different cores. In Figure 5.15, a sketch of the present problem is depicted. On top of the image, a square discretized surface is shown with its two different distributions for each side. Each partition is labeled with R_i, the name of its partition owner core, also known as rank.

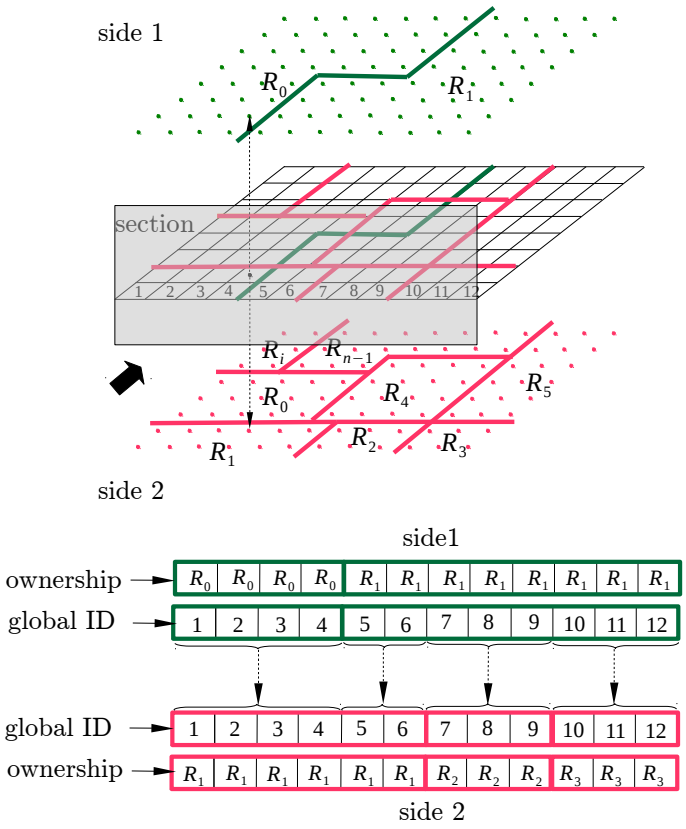

Figure 5.15: Top: Interface surface with different partition patterns for each side. Bottom: Double distribution container diagram for each side, including the global ID for each discrete element and its process ownership. The diagram only includes the elements of the cross-section displayed on the top figure.

An algorithm that makes use of a double distributed data container has been developed for the problem described above. A data container is a storage object associated to a specific topology. It has as many storage positions as nodes are in the topology. In case of parallel execution, the storage positions are distributed among the different cores according to its associated topology partition. In general, if the topology features halo nodes, their positions would also be created to allow the information exchange between processes. Each container partition can only be accessed by its owner process, and therefore, a halo update is required to access data

from the neighboring subdomains. Nevertheless, for a double distributed container, the communications are performed between two different topologies instead of between different partitions of a single topology, and thus, no halos are present.

Before proceeding with the methodology description, the local and global concepts referred to an object or variable will be defined. A local object or variable is defined in all parallel processes, but it may have different sizes and contents depending on the requirements of each particular partition. The data contained in the object is only accessible from its owner core and no information about the object's data in other cores is available. On the other hand, a global object also exists in all cores, but it contains the same data for all parallel partitions. Each core can only access the object copy stored in its memory, but as the content is identical for all cores, it has the same effect as having a single object that can be accessed simultaneously from all parallel partitions.

The double distributed container object is formed by two identical parallel containers but distributed according to two different topologies. These topologies feature no halo nodes since in this case, the information flow is not within the topology distribution itself, but between two different topologies. At the bottom of Figure 5.15, this concept is sketched for the surface section shown above in which, for the sake of clarity, only the processes from R_0 to R_3 and the global IDs from 1 to 12 have been taken into account. If the local containers at both sides were expressed as a global object, they would be identical. However, these vectors are differently distributed in local containers according to their corresponding topology.

The local containers at both sides have to be read and written from their respective partitions. The working procedure to transfer data from one side to the other would work as follows: once the container has been written from one side, a function transfers the data to the container at the opposite side, which now can be accessed from the reading partition. The communication function is bidirectional, being able to transfer data from side 1 to side 2 and vice versa.

Regarding the communications, when a pair of linked nodes belonging to different sides are owned by the same processor, the information can be transferred straightforwardly from one local container to the other. This particular situation is represented by the global ID elements 5 and 6 in Figure 5.15, which are owned by R_1 at both sides. On the other hand, when the two linked nodes are owned by two different processes, a communication between two different cores is required to transfer the information from one side to the other.

5.3.2 Algorithm description

The setup and communication algorithms to transfer data from side 1 to side 2 are explained hereafter. The implementation of the reverse procedure is identical, but the other way round. Firstly, a preprocessing step is carried out to generate the necessary

data to perform the communication process successfully. Then, the communication process itself is carried out by executing MPI functions according to the data generated in the initial step. The preprocessing and communication algorithms are summarized below. Then, each step of the algorithms is developed in more detail.

Table 5.2: Communication setup process algorithm.

Algorithm 1: Setup process. Communication data gathering.	
1:	**both sides:** create a global object containing data about the core ownership for each surface element global ID.
2:	**side 1:** create a sending buffer where the data to be send to side 2 will be stored and packed.
3:	**side 2:** create a receiving buffer where the data sent by side 1 will be received by side 2.
4:	**side 1:** create auxiliary buffers allowing the transfer of data from the local container to the local sending buffer.
5:	**side 2:** create auxiliary buffers to unpack and transfer data from the local receiving buffer to the local container.
6:	**side 2:** create auxiliary buffers to read data in side 1 container for the elements owned by the same core at both sides.

Table 5.3: Communication process algorithm. np stands for total number of processors. **Isend** and **Irecv** are standard MPI functions.

Algorithm 2: Communication process.	
1:	**side 1:** information is transferred from local containers to sending buffer following instructions generated in Algorithm 1.
2:	**side 1: for** $0 \leq i < np$ **do Isend** $\rightarrow$ Each core executes an Isend command for each destination process.
3:	**side 2: for** $0 \leq i < np$ **do Irecv** $\rightarrow$ Each core executes an Irecv command for each sending process.
4:	**side 2:** information is transferred from receiving buffer to side 2 local containers following instructions generated in Algorithm 1.
5:	**side 2:** information is directly transferred from side 1 to side 2 local containers for gids owned by the same process in both sides according to instructions generated in Algorithm 1.

Algorithm 1 $\rightarrow$ step 1: The purpose of this step is to allow each partition from both sides to know to which processes is connected to in the opposite side. To do so, each side will generate two global vectors containing the following information: the first vector will contain all global IDs of the whole interface surface. Then, a second vector of the same size will be created with the core ownership data. The information of both vectors is related through the vector position, i.e., the global ID data and its corresponding ownership information are located at the same position for both global vectors. In Figure 5.15, the global vectors that would be generated for the section shown at the top of the figure are displayed. Although in Figure 5.15 the topology partitions are represented, the global vectors would contain all the data and would be accessible for any process at both sides.

Thus, all processes in side 1 will know in which process in side 2 are located its owned global IDs, and therefore, to whom information has to be sent. Conversely, all

processes in side 2 will be able to determine from which processes in side 1 they will receive data and the amount of it.

To generate the global vectors, each process will collect its owned data and then, an **Allgatherv** MPI function is executed to create a copy of the global vector at each core.

Algorithm 1 → step 2: In this step, the side 1 will create a structure to pack the data that has to be sent to other processes. Each partition will generate a vector with the number of total cores involved in the communication. Each position will contain a sub-vector with the data that has to be sent for the local core in side 1, to the corresponding core at side 2. In Figure 5.16, the local sending buffers generated by all section cores in the example of Figure 5.15, are displayed.

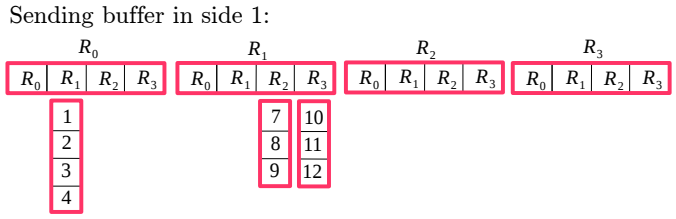

Figure 5.16: Sending buffer structure in side 1.

As an example, the process R_0, which owns global ID 1,2,3 and 4 in side 1, has to send these positions to R_1, which is the owner of these global IDs in side 2. It is worth to point out that global IDs 5 and 6 are not involved in the communication process since they belong to R_1 in both sides, and thus, the information can be transferred straightforwardly without needing a communication step.

Algorithm 1 → step 3: Analogously, each partition in side 2 will create a set of vectors allowing to receive data from side 1 called receiving buffers. The structure is the same as for the sending buffer one, but containing data about the sending processes in side 1. For the example in Figure 5.15, the receiving buffer would be as shown in Figure 5.17.

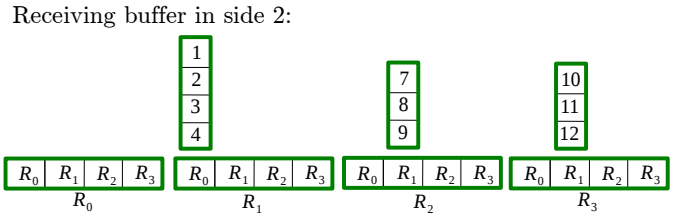

Figure 5.17: Receiving buffer structure in side 2.

Algorithm 1 → step 4: In this step, the data allowing the side 1 topology to transfer data from the local containers to the sending buffer are generated. A couple of vectors are created by each partition. The first one contains the IDs of the elements to be sent to side 2. The second vector stores the destination core identifier for each IDs in the first vector. This information is sufficient to fill the sending buffer.

Algorithm 1 → steps 5 and 6: Steps 5 and 6 are carried out simultaneously. Two different types of data are generated. Firstly, the guides to transfer data from the receiving buffer to the container for the global IDs requiring communication are created. Then, the data buffers allowing a direct transfer of data between containers for global IDs that belong to the same core at both sides are generated.

Both operations are performed at once. Two vectors with the size of the local receiving container are created. The first vector contains boolean variables with a **true** value in case that the corresponding global ID in side 1 is owned by the same core as in side 2. Otherwise, the value is **false**. The second vector contains different data depending on the boolean variable value in vector 1. If for a given position, the first vector value is **true**, in vector 2, the ID of the corresponding position of the container in side 1 is stored. Thus, the receiving container knows from which position in the side 1 container can directly read the data. On the other hand, if the value in vector 1 is **false**, in vector 2, the sending core identifier is available such that the container in side 2 can access the corresponding receiving buffer position.

At this point, all the data allowing the communication between the two interface sides has been generated. Hereafter, the necessary steps for a communication process summarized in Algorithm 2 are developed:

Algorithm 2 → step 1: Firstly, the sending side transfers data from its local containers to the sending buffer according to the instructions generated in step 4 of Algorithm 1.

Algorithm 2 → step 2: In this step, the information is sent from side 1 to side 2 for those global IDs that require communication by using appropriate MPI functions. The sending process is carried out through an **Isend** nonblocking function. This function prevents from reusing the sending buffer before a communication success message is received. However, it is not a synchronous sending function, which would block any new sending process before the message has been received by the destination core.

The sending procedure is executed as follows: Each partition in side 1 executes the **Isend** function for each destination core i at side 2. The data to be sent is the vector in the local sending buffer corresponding to the destination core i:

step 2:

---

for $0 \leq i < np$ **do**{
  Isend(i,localSendingBuffer[i]);
}

Algorithm 2 → step 3: In this step, the receiving **Irecv** MPI function is executed from side 2. Each partition in side 2 executes the **Irecv** function for each sending core i at side 1. The incoming data from a given sending process i will be stored in the local receiving buffer according to its core identifier:

step 3:

---

for $0 \leq i < np$ **do**{
  Irecv(i,localReceivingBuffer[i]);
}

As an example, the size of the buffer vector sent by core R_0 to R_1 (Figures 5.16), matches the size of the receiving buffer vector in core R_1 for the sending core R_0 (see Figure 5.17).

Algorithm 2 → steps 4 and 5: These two last steps are performed simultaneously. The data generated in steps 5 and 6 in Algorithm 1 are used to fill the local receiving containers. The information is taken directly from the container in side 1 or from the receiving buffer according to the boolean vector predefined in Algorithm 1, steps 5 and 6.

5.3.3 Wall shear stress evaluation

The wall model shear stress is evaluated from the accurate near-wall velocity profile computed in the TLM grid. In Figure 5.18, the discrete velocity profile right over a solid wall face is sketched. The orange points are the TLM mesh nodes generated by the extrusion of the solid surface along the normal direction. For a continuous velocity field, the wall shear stress in a given wall-parallel direction, i, would be evaluated as:

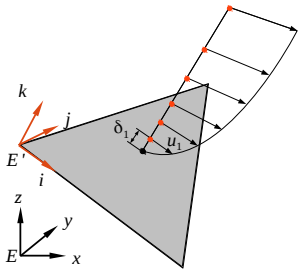

Figure 5.18: Solid face with its local coordinate axis $\{i, j, k\}$ and the velocity vector field in the wall-normal direction.

$$\tau_i = \mu \frac{\partial u_i}{\partial k}\bigg|_{k=0}, \tag{5.3}$$

where k is the solid face normal direction. Since the velocity derivative term in Equation 5.3 is evaluated at the wall $(k = 0)$, for the discrete form of the equation, the first off-wall velocity value (u_{1i}) is used:

$$\tau_i \approx \mu \frac{u_{1i}}{\delta_1}, \tag{5.4}$$

The output of the TLM model is the wall shear (parallel) stress referred to the global reference frame $(\boldsymbol{E} = \{\boldsymbol{x}, \boldsymbol{y}, \boldsymbol{z}\})$, $\boldsymbol{\tau}_{xyz} = \tau_x \boldsymbol{x} + \tau_y \boldsymbol{y} + \tau_z \boldsymbol{z}$. Given that the flow variables are referred to the global frame $\boldsymbol{E}$, to obtain the wall-parallel component of the shear stress, a change of basis of the velocity field has to be carried out from the global frame to a local basis parallel to each solid face $(\boldsymbol{E'} = \{\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}\})$. This local reference frame $\boldsymbol{E'}$ is generated for each solid face, and it is constructed by using a face edge as a vector $\boldsymbol{i}$, the face normal vector as $\boldsymbol{k}$, and the second parallel component is derived through the cross product of the two previous unit vectors $(\boldsymbol{j} = \boldsymbol{k} \times \boldsymbol{i})$. The transformation matrix for a change of basis from $\boldsymbol{E}$ to $\boldsymbol{E'}$ is $\boldsymbol{T}$, while its inverse $\boldsymbol{T}^{-1}$, is a linear transformation from $\boldsymbol{E'}$ to $\boldsymbol{E}$.

$$\mathbf{u}_{xyz} = u_x \boldsymbol{x} + u_y \boldsymbol{y} + u_z \boldsymbol{z} \quad \xrightarrow{\mathbf{u}_{ijk} = \boldsymbol{T}\mathbf{u}_{xyz}} \quad \mathbf{u}_{ijk} = u_i \boldsymbol{i} + u_j \boldsymbol{j} + u_k \boldsymbol{k}, \tag{5.5}$$

where $\boldsymbol{u}$ is the velocity field. Once the wall-parallel velocity components u_i and u_j are obtained, the wall shear stresses, τ_i and τ_j, can be evaluated from Equation 5.4. Then, the wall-parallel components referred to the $\boldsymbol{E'}$ frame have to be transformed back to the global basis $\boldsymbol{E}$ to obtain the final model output, τ_x, τ_y and τ_z:

$$\boldsymbol{\tau}_{ijk} = \tau_i \boldsymbol{i} + \tau_j \boldsymbol{j} + 0\,\boldsymbol{k} \quad \xrightarrow{\boldsymbol{\tau}_{xyz} = \boldsymbol{T}^{-1}\boldsymbol{\tau}_{ijk}} \quad \boldsymbol{\tau}_{xyz} = \tau_x \boldsymbol{x} + \tau_y \boldsymbol{y} + \tau_z \boldsymbol{z}, \tag{5.6}$$

5.3.4 Wall shear stress boundary condition for the LES domain

Once the wall shear stress has been evaluated by the model and transferred to the LES partition, the new values have to be introduced to the LES computation as a boundary condition. It has to be taken into account that the methodologies described in this section are applicable to any wall shear stress model.

During the TLM development process, three different strategies were implemented and tested. The source term approach explained below was finally adopted, mainly because of numerical stability reasons and code simplicity.

The application of an external shear stress to a cell solid face basically affects the diffusive flux at this face. Therefore, a methodology intended to modify the LES diffusive term at this specific face must be developed.

First of all, we recall how the diffusive flux is evaluated at the LES cell faces according to the second-order central-difference numerical schemes used by TermoFluids. Taking the diffusive term from the incompressible filtered NS equations (See Chapter 1) and applying the finite volume method, we obtain the following discretization:

$$\begin{aligned} D_i = \oint_\Omega \nabla \cdot (\mu + \mu_{sgs}) \nabla \boldsymbol{u}\, d\Omega = \oint_S (\mu + \mu_{sgs}) \nabla \boldsymbol{u}\, \boldsymbol{n} dS \approx \\ \approx \sum_{f=1}^{m} (\mu + \mu_{sgs})_f \left[\frac{u_{iNB} - u_{iP}}{\delta_n} \right] S_f, \end{aligned} \tag{5.7}$$

The discrete diffusive term for the LES equations, D_i, is expressed as a summation of diffusive fluxes at the cell faces, D_{f_i}. In case of a boundary face, in which the neighbor node, NB, is the face node itself, the diffusive flux might be expressed as:

$$D_{f_i} = (\mu + \mu_{sgs})_f \left[\frac{u_{iF} - u_{iP}}{\delta_n} \right] S_f, \tag{5.8}$$

where u_{iF} is the velocity at the solid face. Since this boundary node by definition is of solid wall type, this value should always be zero.

According to Equation 5.8, there are several ways in which the diffusive flux, D_{f_i}, can be modified. In our case, we have considered three approaches. The first one is based on modifying the velocity gradient value by altering the fluid velocity at the wall. Another approach consists in changing the LES subgrid viscosity (μ_{sgs}) at the wall, and finally, the replacement of the whole term value for a new one is considered. These three options lead to three different implementations that are hereafter detailed.

Slip wall boundary conditions

In this approach, the fluid velocity at the solid faces will be replaced by a non-zero value. In order to obtain the appropriate velocity at the LES wall node, we will require the diffusive term at the LES face, D_{f_i}, to be equal to the diffusive flux provided by the wall shear stress model, $\tau_i S_f$:

$$\tau_i S_f = D_{f_i} = (\mu + \mu_{sgs})_f \left[\frac{u_{iF} - u_{iP}}{\delta_n} \right] S_f \rightarrow u_{iF} = \frac{\tau_i \delta_n}{(\mu + \mu_{sgs})_f} + u_{iP}, \tag{5.9}$$

A new slip wall boundary condition was created for the LES domain which used the wall shear stress of the model to evaluate the slip velocity through Equation 5.10. However, this approach was finally dismissed because in geometries with sharp edges, it could generate some instabilities. Additionally, from a physical point of view, introducing a non-zero velocity at the wall is an artificial condition that we preferred to avoid.

Subgrid viscosity approach

Another possible way to modify the diffusive flux at the wall is by modifying the subgrid viscosity at the wall. For a physically consistent LES model, the behavior of the subgrid viscosity, μ_{sgs}, in the wall vicinity should be of $\mathcal{O}(y^3)$, vanishing at $y = 0$.

This approach could be seen as if the wall model was integrated into the subgrid model itself. Applying a non-zero μ_{sgs} value at the wall could be interpreted as the LES model having a specific wall behavior to account for underresolved boundary layer effects. However, from a conceptual point of view, we preferred not to introduce unphysical conditions.

The new value of μ_{sgs} at the wall would be obtained the same way as in the slip wall method, i.e., by requiring the LES diffusive term to be equal to the diffusive value provided by the wall model:

$$\tau_i S_f = D_{f_i} = (\mu + \mu_{sgs})_f \left[\frac{u_{iF} - u_{iP}}{\delta_n} \right] S_f \rightarrow \mu_{sgs_f} = \frac{\tau_i \delta_n}{u_{iF} - u_{iP}} - \mu_f, \tag{5.10}$$

Source term method

The source term approach was the finally adopted implementation. It is numerically stable, easy to implement, and in our opinion, a natural way to introduce an external effect to the filtered NS equations from a physically-consistent point of view.

The strategy was based on adding a source term, $\mathcal{S}_i$, to the LES equations only affecting the wall adjacent cells. The idea is to subtract the diffusive contribution of

the solid face evaluated by the LES, D_{f_i}, to the cell total term D_i. Then, the diffusive flux provided by the wall model is added to the equation. Mathematically, this is expressed as:

$$\mathcal{S}_i = -D_{f_i} + \tau_i S_f = -(\mu + \mu_{sgs})_f \left[\frac{u_{iF} - u_{iP}}{\delta_n} \right] S_f + \tau_i S_f, \tag{5.11}$$

This source term can be seen as a correction of the diffusive term. The corrective value is the difference between the diffusive flux at the solid face evaluated by the LES, and the one supplied by the wall model. The physical interpretation of this value is that S_i, is the correction of the wall shear stress deficit caused by a poor near-wall grid resolution of the LES mesh, mainly, in the wall-normal direction.

5.4 Evaluation of parallel efficiency

Once the algorithms and parallel strategy have been developed, an evaluation of the implementation efficiency regarding the computational resources usage will be carried out. To do so, a standard strong-scalability test will be performed. This test analyzes how a given algorithm is able to leverage the available computational resources as the number of workload partitions increases. The test is based on a sequential computation with a maximum size limited by the single core available memory. Then, the workload is subsequently subdivided by powers of 2 such that the load per core is halved at each step while keeping the global test size. Ideally, as the number of cores is doubled, the required computational time should be divided by two. Under ideal circumstances the algorithm speedup (the ratio between sequential and parallel work units per time unit) scales linearly. However, in practice, this does not happen. In parallel execution that uses domain decomposition, halo nodes are generated as communications between processes are created. This affects the code efficiency to the extent that, for a sufficiently large number of cores, the communication costs will overtake the gains obtained by reducing the workload per core. From this point onwards, increasing the number of partitions would entail a rising in the total computational time. In general, this situation happens when the workload per core is so low that the benefits of reducing it are negligible, while a further division entails a communication burden increase.

This kind of scalability test is suitable to evaluate the performance of codes intended for long-time running computations. In this case, the main goal is finding a compromise between the number of used cores allowing the computation to complete in a reasonable amount of time, while not wasting too many resources due to parallel overhead derived from additional communication requirements. In general, CFD computations are extremely demanding from both, time and memory points of view.

In this work, the efficiency has been analyzed from a running time perspective. However, to analyze the algorithm performance with an efficient memory usage criterion, a weak scalability test should be performed.

In order to analyze the efficiency enhancements introduced by the present parallelization strategy, its speedup performance will be compared with the one of the trivial parallelization approach. It is worth to recall that in the latter strategy, only the processors of the LES partition owning a portion of the solid wall are involved in the wall model resolution.

Additionally, the influence of the solid boundary extent with respect to the global computational domain size will be analyzed. To do so, additional tests with two different meshes will be performed. These two meshes will feature the same total amount of inner nodes, but with different solid boundary sizes with a factor of two.

In order to ensure that the flow physics do not affect the scalability tests and that only algorithmic efficiency is being analyzed, the iterative solvers involved in the computations are required to perform the same number of iterations at each solving process.

5.4.1 Tests definition

The tests have been carried out at the in house JFF supercomputing cluster which features 40 nodes. Each node has 2 AMD Opteron with 16 Cores for each CPU linked with 64 Gigabytes of RAM memory and an infiniband QDR 4X network interconnection between nodes with latencies of 1.07 microseconds with a 40Gbits/s bandwith.

As previously commented, the size of a strong scalability test is limited by the memory available for a single core. An individual processor can use exclusively all the memory available in its node, and therefore, the memory limit for our tests is 64Gb. This allows a 3.2 million control volumes mesh test.

The computational domain for the tests is box-shaped. It has been discretized according to two different partitions, $n_x, n_y, n_z = 560 \times 280 \times 20$ and $560 \times 140 \times 40$, where n_x, n_y, and n_z, are the number of nodes in the x, y, z directions, respectively. Both partitions feature the same number of grid points (3.1×10^6) but with different bottom surface sizes of 156×10^3 and 78×10^3, respectively. From here onwards, the mesh with a refined bottom surface will be named M1, while the grid with a coarse surface discretization will be called M2. For the TLM mesh, the bottom surface has been extruded in five layers for all tests. The four different speedup test configurations are summarized in Table 5.4:

Table 5.4: Definition of the speedup tests for the TLM parallel implementation.

Test	Parallel scheme	Mesh	LES mesh size	TLM mesh size
1	TLM partition	M1: $560 \times 280 \times 20$	(3.1×10^6)	(7.84×10^5)
2	Trivial	M1: $560 \times 280 \times 20$	(3.1×10^6)	(7.84×10^5)
3	TLM partition	M2: $560 \times 140 \times 40$	(3.1×10^6)	(3.92×10^5)
4	Trivial	M2: $560 \times 140 \times 40$	(3.1×10^6)	(3.92×10^5)

Regarding the number of parallel partitions, all tests have been carried out from the sequential configuration (core 1), up to 256 CPUs, doubling the number of cores at each subsequent test.

5.4.2 Tests results

The results of the speedup tests are presented in Figures 5.19 and 5.20 for mesh M1 (tests 1 and 2) and M2 (tests 3 and 4), respectively. These charts represent the algorithm efficiency vs. the number of parallel partitions. The efficiency is measured as the inverse of the computational time, such that, the higher the time the lower the efficiency. The efficiency results of the two TLM partition methods are presented together with the speedup curve of an LES-only computation (i.e., without wall model). This allows assessing the TLM algorithm performance with respect to the LES implementation design. Since we are comparing different algorithms (LES, TLM with trivial and specific partitions) and we are only interested in evaluating their efficiency in terms of resource usage, the computational times for each algorithm have been non-dimensionalized. The sequential computational time has been used as a reference such that the sequential efficiency is 1 for all algorithms.

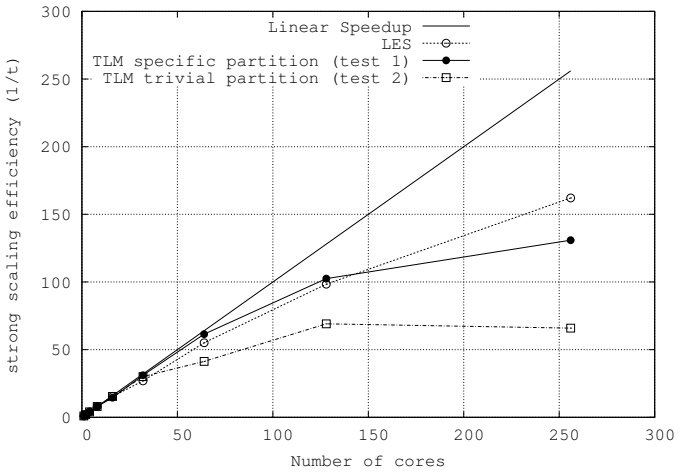

Figure 5.19: Speedup tests for mesh M1. See Table 5.4 for details.

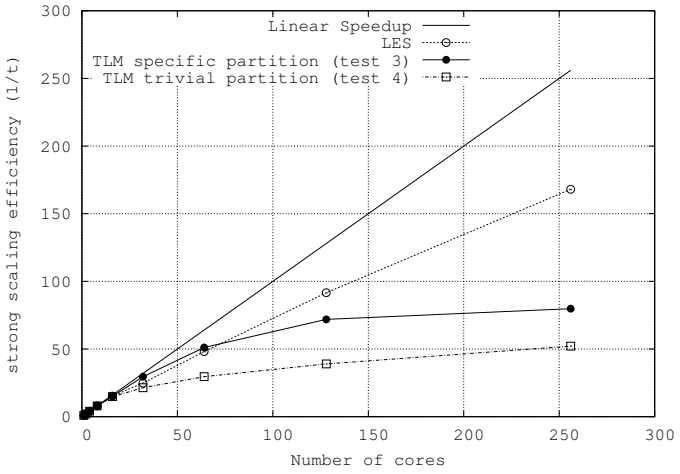

Figure 5.20: Speedup tests for mesh M2. See Table 5.4 for details.

From the speedup plots it can be observed that, as expected, all algorithms scale linearly (maximum efficiency) for a small number of partitions (i.e., up to 16). Then, efficiency starts to decrease due to the increased communication costs. However, this

behavior is not uniform for all algorithms and meshes. While the LES speedup curve is almost independent of the mesh, the TLM performance strongly depends on this factor. This should be expected since both LES meshes (M1 and M2) have the same number of control volumes whereas, the TLM mesh size in tests 1 and 2, is the double than in tests 3 and 4. For the large TLM mesh (M1), the efficient TLM parallelization strategy (test 1) performs especially well, scaling linearly up to 64 CPUs, even better than the LES-only behavior. Then, its performance starts to decay, ending up slightly below the LES one at 256 CPUs. On the other hand, the trivial parallelization strategy performs significantly worse than the rest. The behavior of this strategy is analyzed in more detail further in this section.

In order to explain the speedup behavior of the efficiently parallelized TLM, it has to be taken into account that, in general, the largest portion of memory is dedicated to the LES domain, being the limiting factor of a given simulation. Thus, the decision about the optimal number of parallel partitions is made considering the LES domain constraints. Assuming an efficient parallelization strategy for both domains, if the TLM total workload is similar to the LES one, the point at which both speedup curves peak, will be similar. On the other hand, if the workload of the TLM is significantly lower than the LES one, it has to be expected that the TLM curve peak will move towards smaller partition numbers. This is because the inefficient situation, in which an extremely low workload per core is combined with high communication costs, is reached much before in the TLM partition than in the LES domain. This behavior can be observed by comparing test 1 and test 3 curves. Both follow the same parallelization strategy but with a halved total workload for test 3. As can be observed, the speedup curve decay starts much earlier for test 3, in which the total workload is significantly lower. This analysis cannot be extrapolated for inefficient parallel implementations (test 2 and 4) given that other factors may affect the speedup curve behavior. Those factors are commented below.

Another aspect that may affect the performance of the TLM parallel strategy with respect to the LES one is the way the mesh is partitioned. As commented at the beginning of this chapter, the Metis algorithm is used for partitioning the mesh associated graph, such that a balanced workload distribution is obtained while minimizing the communication costs. While for the LES domain this strategy is applied to the total 3D mesh, in the TLM domain, the Metis algorithm is applied to the superficial mesh rather than the extruded 3D mesh, whose optimal partitions are not necessarily equal. This is especially significant for small solid surfaces and large number of extrusion layers. In this case, a wall-parallel partition would entail a small number of communications, while for a wall-normal partition (current implementation), the number of communications for each graph edge partition is equal to the number of extrusion layers. This effect becomes stronger as the number of cores rises since, for a large number of partitions, the weight of communications on the total computational

cost is more significant. This issue may have affected test 1 and 2 efficiency curves. While performing better than their LES equivalent at a low number of cores, they suffer a stronger performance decrease at higher partition numbers.

Another interesting analysis is the comparison between the trivial and the enhanced parallel strategies. Regardless of the TLM mesh size, the efficient parallelization (tests 1 and 3) performs much better than the trivial parallel scheme (tests 2 and 4) at a high number of partitions. This bad behavior of the trivial parallelization strategy is not due to the increase in the number of communications since most of the cores are not involved in the TLM resolution. Instead, it is due to a massive imbalance in the workload distribution among processes, causing that some cores have to assume up to six times the optimal workload while others are idle.

In Figure 5.21, the workload imbalances generated by the trivial parallelization approach are shown. The charts display the number of cores in % with respect to the total number vs. core workload. In the horizontal axis, the nondimensional workload is displayed. The unity represents the optimal workload,i.e., the assignment that a given process would receive with a uniform distribution, while a value of n means that the specified cores have a workload n times higher than the optimal one. On the other hand, in the vertical axis, the percentage of the total available cores with a given workload is shown. The charts are arranged as follows: On moving from top to bottom, the number of mesh partitions decreases from 256 to 8, while from left to right, the TLM mesh grid size increases (M2 left and M1 right). For the sake of clarity, the balances of the less partitioned tests (1,2 and 4) are not displayed given that they showed the same behavior than the 8-core distribution.

With the efficient parallel strategy, we would obtain a distribution chart identical to the one at the bottom right corner of Figure 5.21 regardless of the number of partitions. That means that the 100% of available cores assume the optimal workload. By contrast, the trivial parallelization scheme behaves significantly different as the number of partitions rises. Actually, for both meshes, the workload distribution is reasonably good up to 16 cores, which explains why all the speedup curves scale linearly up to this point. Nonetheless, the smaller TLM mesh (M2) suffers a significant imbalance at 32 cores, explaining why the performance curve of test 4 suffers a significant drop at this point, while for the larger mesh (M1), the efficiency keeps reasonably good up to 64 cores (test 2). This is because the TLM mesh is partitioned according to the LES requirements instead of the TLM ones. Therefore, the smaller the solid surface and the higher the number of partitions, the lower the probability that a given LES partition owns a solid boundary portion. This is well reflected in the distribution charts, in which the percentage of idle cores is significantly higher for the small (M2) than for the large TLM (M1) mesh.

Another undesired effect of the trivial parallelization strategy is not the number of idle cores but the core with the largest assigned workload. Since in a parallel

computation the slowest core determines the speed of the whole, having cores with an excessive workload with respect to the optimal one considerably undermines the global computational performance. This is the reason for the speedup performance to degrade in parallel with the rise of the workload dispersion. For instance, the maximum core workload for M2 is around 6.2, while for M1 is 4.8. This issue adds up to the idle cores problem, explaining the poor performance of test 4.

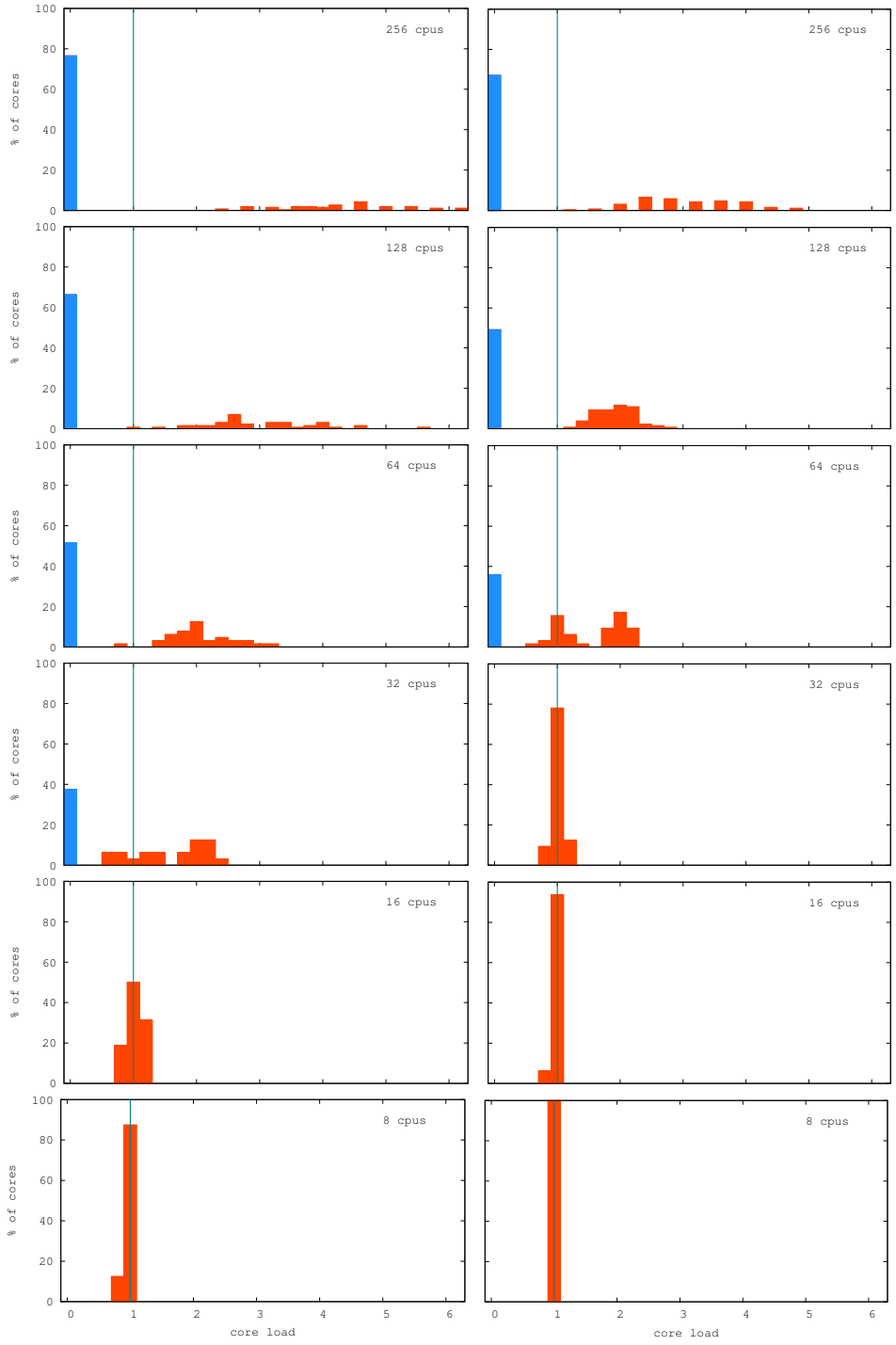

Figure 5.21: Trivial parallelization workload distribution. Percentage of total cores vs. assigned workload. Column in blue corresponds to idle cores. The workload is normalized with the optimal distribution (uniform). Left: M2 results, Right: M1 results. See Table 5.4 for details.

Finally, the performance gains of the efficient parallelization strategy vs. the trivial one in terms of real consumed time will be shown. In Figures 5.22 and 5.23, the inverse of absolute time consumed by the TLM model vs. the number of cores is shown. For the large TLM mesh, it can be observed that at 256 cores, the computational time is more than halved for the TLM specific partition. On the other hand, for small partition numbers, the consumed time is almost identical up to 32 CPUs which is coherent with the workload balance charts presented above. It could be argued that for small partitions, the costs of the specific TLM partition should be higher than for the trivial one since the former has the additional burden of communicating data through the top and bottom interfaces. By contrast, the trivial approach can exchange information with the LES domain straightforwardly. However, this is not the case since the costs of the LES/TLM interface communications are almost negligible compared to the total TLM computational cost. In the worst case (test 1 with a partition of 64 cores), the TLM/LES communications represent the 7% of the total.

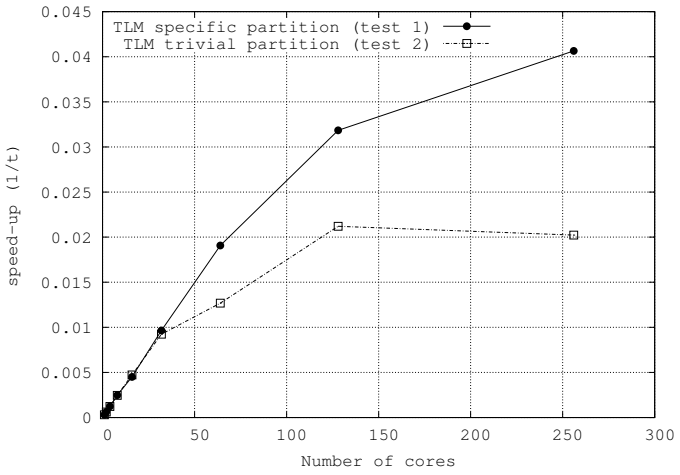

Figure 5.22: Dimensional parallelization efficiency in terms of real consumed time, $[s^{-1}]$, for mesh M1.

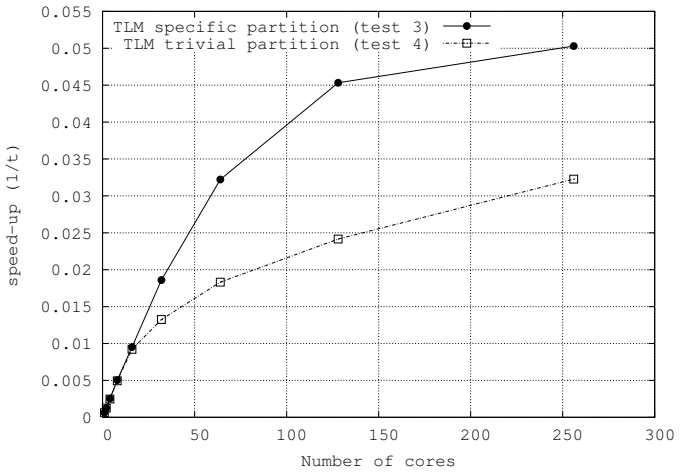

Figure 5.23: Dimensional parallelization efficiency in terms of real consumed time, $[s^{-1}]$, for mesh M2.

5.5 Conclusions

In this Chapter, a strategy allowing to resolve the TLM numerical model of Chapter 4 has been thoroughly explained, including details of algorithm required for the practical interaction between a TLM and an LES computation.

Since the efficient use of computational resources is of capital importance in CFD computations, an adequate parallel methodology has been developed for the TLM. The implementation and the parallel strategy details are exhaustively developed along the text, and finally, an evaluation of the final algorithm performance is carried out.

The parallel method developed for the TLM shows good strong scalability. However, the parallel efficiency decays for small ratios of TLM/LES mesh sizes and a high number of partitions. It has been shown that the larger the solid surface with respect to the LES domain size, the higher the TLM algorithm efficiency. To override this dependency with respect to the solid surface size, a new solution is proposed. The development of a new partition methodology which divided the TLM mesh not only in the wall-normal direction but also in the wall-parallel one would significantly mitigate this undesired effect. Nonetheless, the present parallel approach behaves far better than the initial trivial parallelization scheme in which the TLM mesh partition is performed according to the LES domain requirements. This improvement is observed for any number of processors.

Bibliography

[1] O. Lehmkuhl, C.D. Pérez-Segarra, R. Borrell, M Soria, and A Oliva. Termofluids: A new parallel unstructured cfd code for the simulation of turbulent industrial problems on low cost PC cluster. *Lecture Notes in Computational Science and Engineering*, 67:275–282, 04 2009.

[2] R. Borrell Pol. Parallel algorithms for computational fluid dynamics on ustructured meshes. *Doctoral thesis. Universitat Politècnica de Catalunya*, 2012.

[3] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for paritioning irregualr graphs. *SIAM J. Sci. Comput.*, 20(1):359–392, 1999.

[4] Open MPI, Open Source High Performance Computing. *Web page: https://www.open-mpi.org*.

[5] J. Muela. Modelling and numerical simulation of combustion and multi-phase flows using finite volume methods on unstructured meshes. *Doctoral thesis. Universitat Politècnica de Catalunya*, 2018.

6

Conclusions and future research

6.1 Concluding remarks

In the abstract of this thesis, the necessity of reducing the computational costs of accurate numerical simulations of high Reynolds number wall-bounded flows is remarked. Regarding to such issue, some contributions have been presented.

In the introduction Chapter (1), a new set of expressions intended to estimate the gains of wall modeling compared to explicitly resolving the inner layer are given. Cost estimates were already proposed by Chapman [1] and Choi and Moin[2]. However, the authors only considered the costs related to the grid resolution requirements while the costs associated with time integration were not taken into account. In the present work, the newly proposed expressions comprise both aspects, providing a more realistic estimation for the total computational cost associated with a given high fidelity CFD simulation. According to our estimates, the costs of performing wall-resolved LES compared to wall-modeled LES are even more extreme than those estimated by Chapman and Choi and Moin, strongly highlighting the necessity of general wall modeling strategies.

6.1.1 Analysis tools for high fidelity CFD simulations

In Chapter 2, several tools intended to assess the reliability of a given high fidelity CFD simulation are presented. In this regard, a methodology intended to determine the computational grid resolution suitability for a given DNS or LES simulation is proposed. According to Pope[3], an LES mesh must support flow scales equal to or

larger than those in the dissipation/inertial range limit to avoid significant accuracy losses. The presented approach determines how far a spatial grid resolution is from this limit, allowing to estimate the LES model validity in the far field region. The presented tools, including the grid resolution assessment technique, are further used in Chapter 3 in which a non-equilibrium Two-Layer wall model (TLM) for LES is presented. The tools are used to ensure the correct setup of the TLM validation tests.

6.1.2 Two-Layer wall model for non-equilibrium flows

In Chapter 3, a Two-Layer wall model [4] suitable for non-equilibrium wall flows is presented in order to contribute to seeking solutions for the challenging difficulties commented above. The model is based on the full incompressible URANS equations in order to provide a general model applicable to any flow condition.

Wall shear stress models in general and RANS-based wall models (WM) in particular are affected by the "log-layer mismatch" (LLM) and the resolved Reynolds stresses (RRS) inflow problems which undermine the quality of the WM numerical predictions. Until now, these two problems have been treated separately with different techniques[5, 6, 7], which in some cases were computationally expensive and geometrically complex[5] or they were affected by a lack of generality[6]. Yang *et al.* [7] used the time-averaging strategy to address the LLM problem[7] in the context of a wall function for equilibrium flows. In the present work, this approach has been applied for the first time to a Two-Layer model. For this particular application, the time-average filter (TAF) mission is not only solving the LLM issue, but also mitigating the RRS inflow problem at the same time, with a general and low-computational-cost step. The proposed approach is exceptionally efficient in avoiding the RRS inflow consequences and combined with its effects on the LLM, it dramatically reduces the complexity of the TLM formulation and implementation while strongly increasing its efficiency and geometrical range of applicability compared to other existing TLM strategies[6].

To correctly set up the WM, a methodology to determine the necessary temporal filter length, T, is proposed and validated in equilibrium and non-equilibrium conditions. This approach is based on the analysis of the velocity power spectrum to reveal the flow characteristic time scales. Through this methodology, it is found that when feeding a RANS domain with time-resolved LES data, frequencies higher than those in the energy-containing/inertial range limit must be filtered.

The exisiting approaches intended to deal with the RRS inflow [5, 6] only take into account the apparent diffusive effects of the resolved scales on the RANS-based wall layer. According to our results, this is not sufficient. The inertial subrange scales, although not inherently diffusive, they make the RANS model working out of its range of applicability. In the RANS approach, only the mean flow is solved, and thus, the model expects only mean-flow-based data to deliver solutions. When

time-resolved data is introduced through the boundary, it causes an overprediction of the mean-flow strain, and therefore, an excess of modeled diffusivity.

On the other hand, the filter size influence on large-scale unsteady flow motions is assessed through a strongly unsteady non-equilibrium flow. It is found that for this kind of flows, the effects of the largest motions have to be taken into account since they play an essential role in the global behavior. However, it remains unclear whether the value of the filter length directly influences the largest time-scales, or their modification is an indirect consequence of having a poorly resolved mean flow due to inadequate filtering periods.

Moreover, the model generality from a geometric point of view is validated through the application of the TLM to a fully 3D geometry. Specifically, the flow around an Ahmed car body is computed with satisfactory results. To the author best knowledge, there are no publications of TLM applications in fully 3D complex geometries.

Finally, the interaction of the wall model with the LES model is assessed. It is concluded that the WM acts as an integral part of the LES model, overriding the wall behavior of each particular subgrid model.

6.1.3 Mathematical and numerical approach and its algorithmic implementation

The presented TLM is a fully operational and independent CFD solver. The current implementation automatically generates an unstructured mesh which is partitioned according to its own parallel distribution scheme. Additionally, the TLM solver interacts bidirectionally with an existing LES solver which has an entirely different parallel distribution. While the TLM solver uses time-resolved LES data as a boundary condition, the wall model feeds back the LES domain with a wall shear stress which is derived from the near-wall RANS velocity field.

The mathematical model of the TLM is based on the URANS equations, which are solved numerically through the finite volume method. An implicit time integration scheme is applied to avoid CFL constraints. On the other hand, in order to solve the velocity-pressure coupling, a fractional-step method is used. This projection method features a pressure-correction step in order to minimize its characteristic numerical and splitting errors[8].

An entirely new algorithmic infrastructure has been implemented to allow the resolution of the mathematical and numerical model commented above. Since the primary objective of a TLM is the reduction of the total computational costs, the implementation efficiency is a crucial aspect. Thus, the parallel efficiency of the present code has been tested through a strong scalability test. The present implementation reaches excellent load balancing among processors, and it shows good strong scalabil-

ity up to 128 CPU's. For a larger number of partitions, it shows a weaker performance compared to the LES domain one. This is because, unlike the LES domain, the mesh partition is carried out considering the solid surface topology instead of taking into account the full 3D mesh.

6.2 Future research

A crucial conclusion obtained from the TLM filter length analysis in Chapter 3, is that the necessary value of T is several orders of magnitude larger than the computational LES time-step, Δt. Depending on the test, the value of Δt ranges from approximately $10^{-2}\,TU$ in the high Reynolds number Pipe flow case to $10^{-4}\,TU$ in the airfoil and the Ahmed car tests. By contrast, the TLM filtering periods are of the order of approximately 10^{1}. Since the real temporal variation of the wall shear stress provided by the model is related to the filter length T, this entails that the wall shear stress signal is completely oversampled. Taking advantage of the implicit time integration of the model governing equations, it should be investigated how many TLM iterations could be skipped for each LES iteration without loss of accuracy. Considering that the ratio $T/\Delta t$ ranges between 10^3 and 10^5, it is likely that a very significant number of wall model iterations could be saved. Depending on the results of this analysis, that could entail that we could obtain a general non-equilibrium wall model at a slightly higher cost compared to an equilibrium wall function. In any case, the high values of the $T/\Delta t$ ratio show promising cost-cutting potential for the TLM approach.

On the other hand, as a conclusion of the strong scalability test, further improvements in the TLM efficiency could be obtained by performing a three-dimensional partition of TLM mesh. This would be especially beneficial for cases with a large number of extrusion layers in which the number of halo nodes could be optimized.

Finally, another important conclusion of the present work is that the increase of the time-step upper bound in explicit computations has considerable potential in cutting down the computational costs for high Reynolds number flows. Probably, even more than the reduction of the near-wall spatial resolution requirements since it is very difficult to parallelize the advancement in time. The development of new strategies intended to allow larger time-steps should also be further investigated. The use of variable implicit-explicit time integration schemes is a good candidate to avoid the CFL constraints, while the development of new turbulence models intended to redress the effects of poor temporal resolutions could complement this strategy.

Bibliography

[1] D. R. Chapman. Computational aerodynamics, development and outlook. *AIAA J.*, 17:1293–1313, 1979.

[2] H. Choi and P. Moin. Grid-point requirements for large eddy simulation: Chapman's estimates revisited. *Phys. Fluids*, 24:011702, 2012.

[3] S. Pope. Ten questions concerning the large-eddy simulations of turbulent flows. *New J. Phys.*, 6(35):1–24, 2004.

[4] E. Balaras, C. Benocci, and U. Piomelli. Two-layer approximate boundary conditions for large-eddy simulations. *AIAA J.*, 34 (6):1111–1119, 1996.

[5] S. Kawai and J. Larsson. Wall-modeling in large eddy simulation: Length scales, grid resolution, and accuracy. *Phys. Fluids*, 24:015105, 2012.

[6] G. I. Park and P. Moin. An improved dynamic non-equilibrium wall-model for large eddy simulation. *Phys. Fluids*, 26:015108, 2014.

[7] X. I. A. Yang, G. I. Park, and P. Moin. Log-layer mismatch and modeling of the fluctuating wall stress in wall-modeled large-eddy simulations. *Phys. Rev. Fluids*, 2:104601, 2017.

[8] A. Carmona, O. Lehmkuhl, C. D. Pérez-Segarra, and A. Oliva. Numerical analysis of the transpose diffusive term for viscoplastic-type non-newtonian fluid flows using a collocated variable arrangement. *Numer. Heat Tr. B-Fund.*, 67:410–436, 2015.

Appendix A

The Helmholtz-Hodge Theorem

This appendix presents the Helmholtz-Hodge theorem [**?** **?**] which is the mathematical basis of the Fractional-step method. This method allows the resolution of the velocity and pressure coupling in the incompressible Navier-Stokes equations.

A.1 The Helmholtz-Hodge Theorem

Theorem 1. *A given vector field* $\mathbf{u}$ *defined in a bounded domain* Ω *with smooth boundary* $\partial\Omega$*, is uniquely decomposed in a pure gradient field and a divergence-free vector parallel to* $\partial\Omega$*:*

$$\mathbf{u} = \mathbf{s} + \nabla\varphi \tag{A.1}$$

where

$$\nabla \cdot \mathbf{s} = 0 \qquad \mathbf{s} \in \Omega \tag{A.2}$$
$$\mathbf{s} \cdot \mathbf{n} = 0 \qquad \mathbf{s} \in \partial\Omega \tag{A.3}$$

being $\mathbf{n}$ the normal vector to $\partial\Omega$.

Proof.

we consider the following vector identity derived by applying the chain-rule:

$$\nabla \cdot (\varphi\mathbf{s}) = (\nabla\varphi) \cdot \mathbf{s} + \varphi(\nabla \cdot \mathbf{s}) \tag{A.4}$$

and being **s** a divergence-free vector (See Equation A.2), we obtain:

$$\nabla \cdot (\varphi \mathbf{s}) = (\nabla \varphi) \cdot \mathbf{s} \tag{A.5}$$

Integrating previous expression over the domain Ω and applying the Gauss divergence theorem, it leads to:

$$\oint_\Omega (\nabla \varphi) \cdot \mathbf{s} d\Omega = \oint_\Omega \nabla \cdot (\varphi \mathbf{s}) d\Omega \quad \xrightarrow{\text{Gauss}} \quad \oint_\Omega \nabla \cdot (\varphi \mathbf{s}) d\Omega = \oint_{\partial\Omega} (\varphi \mathbf{s} \cdot \mathbf{n}) dS \tag{A.6}$$

The Boundary integral vanishes if **s** satisfies the prescribed condition in Equation A.3. Therefore, the orthogonality[1] between vector field **s** and $\nabla\varphi$ is proved:

$$\oint_\Omega (\nabla \varphi) \cdot \mathbf{s} d\Omega = 0 \tag{A.7}$$

From the previous expression, the decomposition unicity can be proved. Firstly, it is assumed that two different decompositions do exist:

$$\mathbf{u} = \mathbf{s}_1 + \nabla \varphi_1 = \mathbf{s}_2 + \nabla \varphi_2 \tag{A.8}$$

rearrangin terms:

$$\mathbf{s}_1 - \mathbf{s}_2 + \nabla(\varphi_1 - \varphi_2) = 0 \tag{A.9}$$

By applying the scalar product of previous expression by vector $(\mathbf{s}_1 - \mathbf{s}_2)$ and considering the orthogonality relation A.7, it leads to:

$$0 = \oint_\Omega ||\mathbf{s}_1 - \mathbf{s}_2||^2 d\Omega + \oint_\Omega (\mathbf{s}_1 - \mathbf{s}_2) \cdot \nabla(\varphi_1 - \varphi_2) d\Omega \tag{A.10}$$

$0\ (A.7)$

From equation A.10, it is proved that

$$\mathbf{s}_1 = \mathbf{s}_2 \tag{A.11}$$

Then, from equation A.9, it is also proved that:

$$\nabla \varphi_1 = \nabla \varphi_2 \Rightarrow \varphi_1 = \varphi_2 + C \tag{A.12}$$

It must be pointed out that the unicity of the decmposition (A.1) do not imply, as stated by Equation A.12, that the scalar field φ is uniquely determined.

[1]For the standard inner product of two vector fields **a** and **b**, defined as follows:

$$\langle \mathbf{a}, \mathbf{b} \rangle = \oint_\Omega \mathbf{a} \cdot \mathbf{b} d\Omega \tag{A.13}$$

Appendix B

Main publications in the context of this thesis.

Journal Papers:

- J. Calafell, F.X. Trias, O. Lehmkuhl and A. Oliva. *"A time-average filtering technique to improve the efficiency of Two-Layer Wall Models for large eddy simulation in complex geometries"*. (2019) Accepted in Computers & Fluids Journal.
- D. Volkov and J.Calafell. *A stochastic approach to reconstruction of faults in elastic half space*. (2019) Inverse Problems & Imaging. **13(3)**, pp. 479-511.
- D.E. Aljure, J. Calafell, A. Báez and A. Oliva. *"Flow over a realistic car model: wall modeled large eddy simulations assessment and unsteady effects"*. (2018) Journal of wind engineering and industrial aerodynamics. **174**, pp. 225-240.

International Conference Papers:

- J. Calafell, F.X. Trias and A. Oliva. *"An Efficient Two-Layer Wall Model for High Reynolds Number Large-Eddy Simulation."* 7th European Conference on computational Fluid Dynamics ECCOMAS. Glasgow (UK), 2018.
- J. Calafell, O. Lehmkuhl, I. Rodríguez, A. Oliva and C.D. Pérez-Segarra. *"A Dynamic Wall Model for Larg Eddy Simulation on Unstructured Meshes, Application to Wind Turbine Dedicated Airfoils"*. Turbulence Heat and Mass Transfer 8: Proceedings of the Eighth International Symposium on Turbulence, Heat and Mass Transfer, Sarajevo (Bosnia and Herzegovina), 2015.
- J. Calafell, A. Carmona, O. Lehmkuhl, C.D. Pérez-Segarra and A. Oliva. *"A Dynamic Wall Modelling for Large-Eddy Simulations. Application to High Reynolds Number Aerodynamics."* iTi 2014-Conference on Turbulence. Bertinoro (Italy), 2014.

- J. Calafell, A. Carmona, O. Lehmkuhl, C.D. Pérez-Segarra and A. Oliva. *"Dynamic Wall Modelling for Large-Eddy Simulations. Application to High Reynolds Number Aerodynamics of Complex Geometries."* 6th European Conference on computational Fluid Dynamics ECCOMAS. Barcelona (Spain), 2014.

- J. Calafell, O. Lehmkuhl, A. Carmona, C.D. Pérez-Segarra and A. Oliva. *"A Dynamic Wall Model for Large-Eddy Simulations of Wind Turbine Dedicated Airfoils"*. Journal of Physics: conference series 254. The Science of Making Torque From Wind. Copenhagen (Denmark), 2014.

- J. Calafell, O. Lehmkuhl, I. Rodríguez and A. Oliva. *"On the Large-Eddy Simulation modeling of wind turbine dedicated airfoils at high Reynolds numbers"*. Turbulence, Heat and Mass Transfer 7th symposium. Palermo (Sicily), 2012.

- O. Lehmkuhl, J. Calafell, I. Rodríguez and A. Oliva. *"Large-Eddy Simulations of wind turbine dedicated airfoils at high Reynolds numbers"*. EUROMECH Colloquium 528 Oldenburg (Germany), 2010.

List of Figures

List of Tables