Adaptive Grid Refinement for Hybrid RANS/LES

Sajad Mozaffari

CENTRALE NANTES

THÈSE DE DOCTORAT DE

L'ÉCOLE CENTRALE DE NANTES

ÉCOLE DOCTORALE N° 602
Sciences pour l'ingénieur
Spécialité : « *Mécanique des Milieux Fluides* »

Par
Sajad MOZAFFARI

Adaptive Grid Refinement for Hybrid RANS/LES

Thèse présentée et soutenue à Nantes, le 18 mai 2020
Unité de recherche : UMR 6598, Laboratoire de recherche en Hydrodynamique, Énergétique et Environnement Atmosphérique (LHEEA)

Rapporteurs avant soutenance :

Rickard BENSOW	Professeur	Université Technologie de Chalmers, Suède
Kai SCHNEIDER	Professeur des universités	Université d'Aix-Marseille

Composition du Jury :

Présidente :	Paola CINNELLA	Professeure des universités	École Nationale Supérieure d'Arts et Métiers, Paris
Examinateurs :	Rickard BENSOW	Professeur	Université Technologie de Chalmers, Suède
	Kai SCHNEIDER	Professeur des universités	Université d'Aix-Marseille
	Eric LAMBALLAIS	Professeur des universités	Université de Poitiers
	Marta DE LA LLAVE PLATA	Ingénieure de recherche	ONERA, Châtillon
Directeur de thèse	Michel VISONNEAU	Directeur de recherche	École Centrale de Nantes/CNRS
Co-encadrant de thèse	Jeroen WACKERS	Ingénieur de recherche	École Centrale de Nantes/CNRS

Acknowledgements

Those years of thesis have been a wonderful experience, full of challenges and enriching encounters. I would like to thank the people who helped me during those years.

First, I want to thank my supervisors to give me the chance to work on such an interesting and challenging subject. I would like to thank Dr. Michel Visonneau, for his interesting lectures during CFDIF and PMFLU master courses and his guidance and kind advice throughout my studies, which undoubtedly influenced my decision in choosing this educational path and in particular, for his thoughtful comments and recommendations on this dissertation. I also would like to thank Dr. Jeroen Wackers for his constant support, availability and constructive suggestions, which were determinant for the accomplishment of this work.

Then I thank the members of my committee for investing their time and effort in my work, especially their cooperation during the pandemic. I thank LHEEA for giving me a pleasant working environment. I had the pleasure of being part of the METHRIC team, and I warmly thank its other members: Patrick Queutey, Emmanuel Guilmineau, Alban LEROYER and Ganbo Deng. They have been of great help to me over all these years, and it has been a pleasure to work alongside them.

During this thesis I had the chance to work with brilliant PhD students - now PhDs. I want to thank Yoann for his technical helps especially when I was newcomer and also Catherine for our friendly chats and sharing chocolate.

Last but not least, I am also grateful to my family and other friends who have supported me along the way.

Nantes,
May 2020,
Sajad Mozaffari

Contents

Chapter 1

Introduction

1.1 Turbulence modeling and meshing for realistic flows

Flow modeling of complex, real-life physical phenomena is challenging in terms of computational resources, despite significant developments in computational methods and computing power. Simulation of turbulent flow structures in the wake of a ship, their interaction with waves, the evolution of a wing tip vortex, and the flow detachment behind a car are several real-life examples in which flow structures with different scales exist, which implies that a full simulation of these flows requires to resolve all of these scales. The local error and the accuracy of these simulations are determined by the size of the cells in the mesh on which the governing equations are solved. To obtain satisfactory results that predict accurately the complex physics of the problems, the cells should be sufficiently small (in each of the coordinate directions) especially for the regions where turbulence in the flow is important, or phenomena like discontinuities and large gradients occur. Based on the number of cells that cover the domain of the problem, the number of calculations to be made and thus, the cost of the computation is determined. To resolve all the scales of a realistic turbulent simulation, which is known as Direct Numerical Simulation (DNS), up to 10^{18} cells or more are required. This number of cells is far beyond the order of magnitude that is affordable or even feasible with today's computing technology. Therefore, the application of these computationally costly simulations is limited to only simple flows at low Reynolds numbers.

Turbulence modeling is a way to diminish the computational expenses by decreasing the complexity of the solutions. The idea is to compute a part of the solution using an approximative physical model instead of fully resolving the flow. For turbulence, various models have been introduced and used extensively over the years. All these models constitute a compromise between the computational costs and the accuracy of the solutions (Figure 1.1). For instance, the classic Reynolds-averaged Navier-Stokes (RANS) equations express flow quantities as a sum of mean and fluctuating parts, where the fluctuating part is replaced by an approximate set of equations which is simpler to solve. Since computations with this method are relatively cheap, the simulation of cases with realistic geometries and high Reynolds numbers is feasible. However, the price of this simplicity is losing a part of the physics and therefore, larger differences between the computed solution and reality compared with other approaches. Another class of turbulence models is Large Eddy Simulation (LES), where the large energy containing structures of the flow are directly resolved and only the small structures are approximated by a model. LES improves the quality of the solution compared to RANS. However, while it is still more affordable than DNS, computing costs increase dramatically with respect to RANS, which limits the use of this approach. For example, very high Reynolds number flows or complex geometries are cases where a full LES simulation may not be possible.

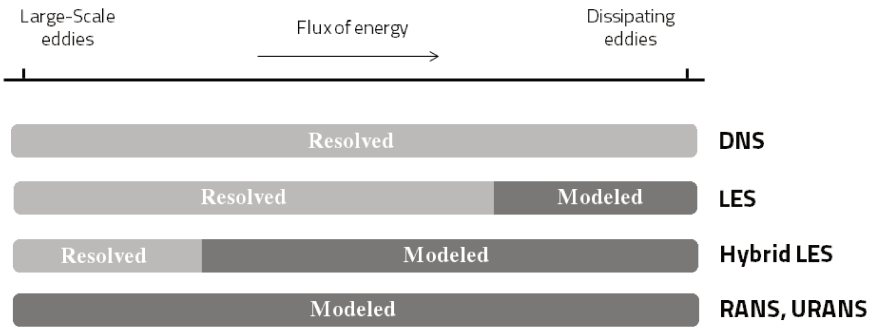


Figure 1.1 – Distribution of resolved and modeled turbulence for different turbulence models

Hybrid RANS/LES models are introduced as a desirable alternative for complicated flows, which inherits advantages of RANS and LES at the same time. A hybrid RANS/LES approach is designed to behave as a RANS model wherever the RANS solution is reliable. For instance, a high Reynolds number attached boundary layer flow is treated correctly by RANS on a mesh with coarse grids at least in one or two directions, while LES generally requires an isotropic mesh of very fine cells inside the entire boundary layer in order to resolve the small-scale flow structures. On the other hand, when RANS is not accurate enough, the LES part of the model is activated. Separated flow is an example of a problem where RANS may lead to a physically wrong solution, while LES predicts a more accurate solution if the grids are locally fine enough. For realistic complex flows, which contain both types of flow features, hybrid RANS/LES turbulence models are effective in reducing computational costs and simultaneously increasing the accuracy.

To take advantage of hybrid RANS/LES models in a real-life problem, a non-uniform mesh with the correct local resolution must be generated. The local properties of the mesh should meet the requirements of the model's behavior. For example, in regions like boundary layers that are treated by RANS, grids can be coarser than in those regions where separation occurs and the model acts as LES. The unnecessary use of small cells imposes an excessive computational cost without any gain and, in opposite, not using small enough cells in regions where LES is activated may deteriorate the LES solution and even in some cases, may lead to completely wrong solutions. Thus, the regions to be covered by fine grids must be identified beforehand. This requires prior knowledge of the global behavior of the flow and the important mechanisms that occur during the simulation, which is impossible for complex geometries. Therefore, to reach an optimal mesh for a hybrid RANS/LES simulation, an iterative construction of the mesh with several complete simulations may be necessary, which requires much time and effort.

Furthermore, in a hybrid RANS/LES simulation, since the local grid resolution is an important factor in determining the behavior of the model, the use of a fixed mesh actually means that the model is forced to treat specific regions only as RANS or only as LES during the entire computation. However, complex real-life flows are often time-dependent and may be composed of macroscopic flow structures which evolve continuously. Thus, changes in the behavior of the model according to the flow in a specific region may be required. The mesh should be constructed to be consistent with the model despite any change in its behavior. This further increases the difficulty of the mesh generation for these types of models.

In these cases, using a mesh adaptation process is an interesting alternative. For adaptive grid refinement (AGR), the mesh is refined automatically and is adapted locally

to the flow features during the computation. As a result, a simulation can start on a coarse mesh with only some local refinement (e.g. to capture the geometry). The number of cells at the beginning of the simulation is relatively low, which saves a lot on computing resources. During simulation, the flow structures are followed by AGR and by locally refining or coarsening grids, an optimal mesh is obtained which only has a high resolution in regions where it is required. The grid modification, depending on the nature of the flow and the capability of the solver, can be performed frequently (every few time steps) or just a few times during the whole computational time. This technique reduces the efforts of the sophisticated and time consuming mesh generation process, especially for complex flows.

1.2 Challenges and objectives

The objective of this thesis is to combine the grid adaptation technique with hybrid RANS/LES simulations and find an adaptive refinement approach which is well-suited for these turbulence models.

Adaptive grid refinement today is used frequently and reliably in RANS simulations. For LES however, any change in the grid properties affects the performance of the model. As is mentioned before, a part of the LES solution is resolved directly and the rest is modeled (Figure 1.1). The amount of each of these two parts depends on the local cell sizes. Thus, modifications of the cell sizes in time by the adaptive refinement change the proportion of the two parts and result in different levels of resolved turbulence before and after the adaptation. In addition, to transfer the solution from the original mesh to the adapted one, interpolation is required. However, due to the strong spatial variation of LES solutions, interpolation errors may be large and cause incorrect representation of resolved turbulence kinetic energy. The same issues also appear at the interface between two regions of the mesh with different grid resolutions as a result of the spatial refinements.

For a hybrid RANS/LES turbulence model, in addition to the high dependency of the LES model on changes in grid resolution by the adaptive grid refinement process, any refinement or coarsening of the grid in a region can switch the model between RANS and LES, regardless of whether those models are capable of predicting the physical phenomena which exist in this region or not. And contrary to RANS simulation where the adaptation is based on an averaged solution which evolves smoothly during computation, for LES and hybrid RANS/LES simulations, intensive unsteadiness in the solutions leads to a significant change in the grid during each adaptation. This raises the possibility of the LES-related issues and leads to frequent switches of the model behavior between RANS and LES during the computation.

The main challenge in this thesis is to find a refinement method which does not suffer from these difficulties. Moreover, the need to treat realistic flows is a challenge for the quality of the adapted mesh, which is also considered in this thesis. On the way to combine the adaptive grid refinement method with hybrid RANS/LES models, first an accurate understanding is sought of the dependency on mesh changes for hybrid RANS/LES models, in particular DES-based models. The possible deviations that occur in the behavior of these models as a result of mesh changes are studied by forcing the turbulence model to function in cases, and on meshes, which are troublesome. Then, to limit the problems due to mesh changes, the choice is made to create adapted meshes which have a more or less static topology with as few changes as possible. Different aspects and choices are considered to find an optimal approach for this type of adaptation.

These studies are carried out using the ISIS-CFD flow solver which was developed by CNRS/ECN with a focus on simulating complex flows in hydrodynamics and is a part of the FINE™/Marine computing suite by NUMECA Int. This unstructured finite-volume based

solver with specific features such as free surface capturing and adaptive refinement enables complex flow computations using RANS and DES-type hybrid RANS/LES turbulence models. The integration of the developed approach in ISIS-CFD will make it possible to use the mesh refinement procedure in hybrid RANS/LES simulations.

1.3 Thesis outline

This thesis starts with two chapters which introduce the models and techniques that are studied. Chapter 2 briefly presents the derivation of the classical RANS and LES turbulence models from the Navier-Stokes equations. It then focuses on hybrid RANS-LES models, in particular the DES-type models and their chronological development. In the last part of the chapter, the main parameters of adaptive grid refinement procedures are introduced and briefly discussed.

Chapter 3 presents a description of the ISIS-CFD flow solver which is used in this thesis. The discretization of mass and momentum conservation equations is presented along with the solution algorithm of the equations. The capabilities of the integrated mesh adaptation process in ISIS-CFD solver are also considered in this chapter.

Chapter 4 is devoted to studying the dependency of the DES turbulence model on the mesh size. A turbulent flow over a flat plate is simulated on static meshes with different mesh sizes. Besides, the performance of the DES model in the presence of mesh size variation is investigated by assessing the spatial evolution of turbulence content across a grid refinement interface in a freely decaying turbulence test case. As a part of this chapter, a synthetic method to generate unsteady velocity fluctuations is described, which is used in these test cases. An early version of this study is described in [68].

In Chapter 5, the time-averaging of the flow features over various averaging intervals and the effect of the averaging method on the averaged solution are considered. Unsteady flow features in a backward-facing step flow are analyzed in an IDDES simulation on a static fine mesh. The determination of the frequency of these instabilities paves the way for studying the effect of different averaging intervals on reducing or removing fluctuations in the averaged solution. This study leads to suitable intervals whose averaged solutions change actively with the flow behavior, while being less dynamic than the instantaneous solution.

In Chapter 6, different refinement approaches based on several time-averaging strategies are tested to obtain an optimal adaptation procedure. The adaptive grid refinement based on averaged solutions is performed in IDDES simulations of a turbulent flow behind a backward-facing step. The time-averaging strategies along with various averaging intervals are investigated in the adaptation process. The impact of these parameters is evaluated by comparing the final adapted meshes and the computational solutions. Parts of this chapter have been published as [69, 70].

Chapter 7 represents the application of the developed average-based AGR to a realistic test case. Simulation of trailing vortices created by sideslip motion of a ship is used to test the capability of the average-based AGR to smoothly capture vortex cores in a RANS and a DES simulation. The solutions are also compared with computations on static fine meshes to verify the applicability of this approach for detailed physical analysis.

Finally, Chapter 8 gives a short summary of the thesis and the conclusions of the performed studies, with suggestions and perspectives for future research.

Chapter 2

Models and techniques

The turbulence problem is a major topic of interest for fluid dynamicists. During decades of research, to solve the governing equations of a turbulent flow and predict the effects of turbulence, different mathematical models were constructed and used. This research field is referred to as turbulence modeling. In the first section of this chapter (Section 2.1), a presentation of the governing equations of turbulent flows is followed by a brief glance at classical RANS and LES turbulence models. In Section 2.2, the development of hybrid RANS/LES turbulence models which benefit from the advantages of both RANS and LES is chronicled.

Along with turbulence models like hybrid RANS/LES closures that keep the balance between the solution accuracy and the computational expenses, the development of more efficient methods to speed up the computations and increase the accuracy of the solutions is of greatest importance. A widely used example is adaptive grid refinement (AGR). AGR is used to optimize the local mesh resolution in the regions where *a priori* unknown flow features are located and avoids excessive refinement where this is not useful. A short description of this method is presented in Section 2.3. The possibility of combining turbulent hybrid RANS/LES simulation with AGR to generate a mesh with sufficient resolution is also discussed in this section.

2.1 Navier-Stokes equations and basic turbulence models

Turbulence is a state of fluid motion, governed by known dynamical laws - the Navier-Stokes equations. For incompressible flow with constant density and viscosity, these are given by:

$$\rho\frac{\partial u_i}{\partial t} + \rho u_j\frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j}, \quad i,j = 1,2,3 \tag{2.1a}$$

$$\frac{\partial u_i}{\partial x_i} = 0. \tag{2.1b}$$

The vectors u_i and x_i are the velocity and position, t is the time, p is the pressure, ρ is the density and τ_{ij} is the viscous stress tensor defined by $\tau_{ij} = 2\mu s_{ij}$. μ is the molecular viscosity and s_{ij} is the strain-rate tensor,

$$s_{ij} = \frac{1}{2}(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}). \tag{2.2}$$

In principle, turbulence is a solution of these governing equations (Eq. 2.1) which is chaotic, spatially and temporally complex. The challenge is to represent the behavior of fluid

motion in a manner suited to the needs of the problem being solved [29]. Solving the three-dimensional, time-dependent Navier–Stokes equations to obtain the chaotic flow field and then averaging the solutions in order to produce statistics, which is referred to as direct numerical simulation (DNS) or fully resolved simulation, is not practicable in most flows of engineering interest. These computations are too costly in terms of computer resources for realistic Reynolds numbers.

Modern strategies for the computation of turbulent flow tend to reduce the dynamical complexity of the Navier-Stokes solutions, while reliably maintaining their main flow phenomena. There are several alternatives to DNS computations to reduce the number of degrees of freedom in the numerical solution. One way is to calculate directly the statistical average of the solution. This approach is called Reynolds-Averaged Navier-Stokes simulation (RANS). In RANS, the use of the averaging concept introduced by O. Reynolds (1895) in which all quantities are expressed as the sum of mean and fluctuating parts, leads to a solution which is cheaper to compute. The total instantaneous velocity $u_i(x,t)$ is decomposed into a sum of its ensemble average $U_i(x,t)$ and a fluctuation part $u_i{}'(x,t)$, such that

$$u_i(x,t) = U_i(x,t) + u_i{}'(x,t). \tag{2.3}$$

The fluctuation $u_i{}'(x,t)$ is not represented directly by the numerical simulation and is included only by way of a turbulence model.

Another approach, which has been in the spotlight in recent decades due to the improvements in computing power, is Large Eddy Simulation (LES). In this approach, only the large scales in space are calculated. The small-scale modes, which are the most computationally expensive to resolve, are removed from the numerical solution via low-pass filtering of the Navier-Stokes equations. The filter, with an associated cutoff length scale Δ which depends on the local grid size, splits any field Φ into a filtered $\overline{\Phi}$ and a sub-filtered or sub-grid scale (SGS) Φ' portion as:

$$\Phi = \overline{\Phi} + \Phi'. \tag{2.4}$$

These assumptions are used to derive modified flow equations for RANS and LES.

2.1.1 Reynolds-Averaged Navier-Stokes simulation (RANS)

The substitution of the decomposition equation (Eq. 2.3) into the Navier-Stokes momentum equation (Eq. 2.1a), and the continuity equation (Eq. 2.1b) followed by ensemble-averaging over these equations yields the following system which is referred to as the Reynolds-averaged Navier-Stokes equations (RANS) [114]:

$$\rho\frac{\partial U_i}{\partial t} + \rho U_j\frac{\partial U_i}{\partial x_j} = -\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j}(2\mu S_{ij} - \rho\overline{u_i'u_j'}), \tag{2.5a}$$

$$\frac{\partial U_i}{\partial x_i} = 0. \tag{2.5b}$$

The quantity $-\rho\overline{u_i'u_j'}$ is an average rate of momentum transfer due to the turbulence which is known as the Reynolds-stress tensor. This symmetric tensor has six independent momentum flux components which are unknown *a priori*. The fundamental problem of RANS is that the equations (2.5) for the averaged turbulent flow are fewer in number than the unknowns. Thus, a prescription for computing $\overline{u_i'u_j'}$ is needed to close the system. This

is where closure modeling helps to formulate further equations such that a solvable set is obtained. Closure models enable one to predict the statistics of turbulent flow by solving a simplified approximate set of equations. During the past decades, different developments of turbulence closures, such as algebraic (zero-equation) models, one-equation models, two-equation models etc. have been proposed.

This thesis does not discuss the history of RANS turbulence modeling or the formulations of the different models. In the following only a short description of the k-ω SST two-equation model, which is the industry standard for hydrodynamics and on which the hybrid RANS/LES model in the ISIS-CFD solver is based, is provided to give readers a general view of the different terms and parameters in the turbulence model that are mentioned in this thesis.

k-ω SST (Shear-Stress Transport) two-equation turbulence model. In this model, in order to improve the prediction of separated flows and to avoid overestimated Reynolds stresses in adverse pressure gradients flows, several desirable elements of different two-equation models are combined. The closure of the Reynolds-averaged equations requires the definition of the turbulent Reynolds stresses, in terms of known quantities in a physically consistent way. In closures based on the Boussinesq hypothesis, the Reynolds stresses for incompressible flows are modeled in terms of an eddy viscosity μ_t as:

$$-\rho\overline{u_i' u_j'} = 2\mu_t S_{ij} - 2\rho k\delta_{ij}/3. \tag{2.6}$$

Therefore, instead of 6 unknowns, only two parameters remain: the turbulent kinetic energy k and the turbulent eddy viscosity μ_t. μ_t can be defined as a function of k and the turbulent frequency ω, as:

$$\nu_t = \frac{\mu_t}{\rho} = \frac{k}{\omega}. \tag{2.7}$$

For the k-ω SST model proposed by Menter et al. [60], this definition of the turbulent eddy viscosity is modified by using a blending function F_2 which is active in boundary layer flows,

$$\nu_t = \frac{\mu_t}{\rho} = \frac{k/\omega}{\max\{1, \Omega F_2/(a_1\omega)\}}. \tag{2.8}$$

Thus, in turbulent boundary layers, the maximum value of the eddy viscosity is limited by forcing the turbulent shear stress to be bounded by the turbulent kinetic energy times $a_1 = 0.31$. This effect is achieved with the absolute value of the vorticity Ω and the blending function F_2, defined as a function of wall distance d:

$$F_2 = tanh\left(\left[\max\left\{2\frac{\sqrt{k}}{0.09d\omega}, \frac{500\mu}{\rho d^2\omega}\right\}\right]^2\right). \tag{2.9}$$

To close the system, two transport model equations for the k and ω scalar turbulence scales are defined as:

$$\rho\frac{\partial k}{\partial t} + \rho U_j\frac{\partial k}{\partial x_j} = \tau_{ij}\frac{\partial U_i}{\partial x_j} - \rho\beta^* k\omega + \frac{\partial}{\partial x_j}[(\mu + \sigma_k\mu_t)\frac{\partial k}{\partial x_j}], \tag{2.10}$$

$$\rho\frac{\partial \omega}{\partial t} + \rho U_j\frac{\partial \omega}{\partial x_j} = \frac{\gamma}{\nu_t}\tau_{ij}\frac{\partial U_i}{\partial x_j} - \rho\beta\omega^2 + \frac{\partial}{\partial x_j}[(\mu + \sigma_\omega\mu_t)\frac{\partial \omega}{\partial x_j}] + 2\rho(1-F_1)\frac{\sigma_{\omega 2}}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}. \tag{2.11}$$

In this formulation, the blending function of the model coefficients (F_1) is equal to one in the inner part of the boundary layer to have Wilcox's k-ω model [114] near solid walls, while in the outer part, it decreases to zero in order to obtain the k-ϵ model [114], in a k-ω formulation, near boundary layer edges and in free-shear layers. The closure coefficients β, β^*, γ, σ_k, and σ_ω are defined by the coefficients of the original k-ω model (ϕ_1) and the transformed k-ϵ model (ϕ_2) as:

$$\phi = F_1\phi_1 + (1 - F_1)\phi_2, \quad \text{where} \quad \phi = \{\beta, \gamma, \sigma_k, \sigma_\omega\}. \tag{2.12}$$

The blending function F_1 is defined as:

$$F_1 = tanh\left(\left[\min\left\{\max\left\{\frac{\sqrt{k}}{0.09d\omega}, \frac{500\mu}{\rho d^2\omega}\right\}, \frac{4\rho\sigma_{\omega 2}k}{C\ D_{k\omega}d^2}\right\}\right]^4\right), \tag{2.13}$$

where $C\ D_{k\omega}$ stands for the cross-diffusion $(\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j})$ in the k-ω model.

2.1.2 Large Eddy Simulation (LES)

As for the RANS turbulence models, the governing equations for LES are also obtained by a decomposition. Substituting the decomposition equation (Eq. 2.4) in the governing Navier-Stokes equation and then filtering the resulting equation gives the equations of motion for the resolved field,

$$\rho\frac{\partial \overline{u}_i}{\partial t} + \rho\frac{\partial}{\partial x_j}(\overline{u}_i\overline{u}_j) = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(2\mu\overline{S}_{ij} - \rho\frac{\partial \tau^r_{ij}}{\partial x_j}\right), \tag{2.14}$$

where all the terms that are not expressed directly from $\overline{u}$ are grouped together in the sub-grid tensor τ^r_{ij} using the Leonard's decomposition [53] over the non-linear term,

$$\tau^r_{ij} = \rho\left(\overline{u_iu_j} - \overline{u}_i\overline{u}_j\right). \tag{2.15}$$

This unclosed term represents interactions among all scales, including filtered scales with unfiltered scales and must be modeled with subgrid-scale (SGS) models. Among several available categories of subgrid-scale turbulence models, the eddy-viscosity model first suggested by Smagorinsky [89] is often used. This model, which is very simple to implement, calculates the SGS stress using the rate-of-strain tensor of the resolved scale $\bar{S}_{ij}$ and the subgrid-scale turbulent viscosity ν_{sgs}. The Smagorinsky model scales the SGS eddy viscosity with the second invariant of the strain rate tensor $|\bar{S}_{ij}|$ and the grid spacing Δ, as [8]:

$$\nu_{sgs} \propto |\bar{S}_{ij}|\Delta^2. \tag{2.16}$$

The term $\bar{S}_{ij}\Delta^2$ was also used by P.R. Spalart when he proposed the first non-zonal hybrid RANS/LES turbulence model (see Section 2.2).

After Smagorinsky, many more complex subgrid models have been proposed and their development still is an active area of research.

2.2 Hybrid RANS/LES

The requirement to directly capture all the scales of motion which are responsible for turbulence production leads to the use of very-fine resolution meshes and increases the

cost of the traditional Large Eddy Simulation method. Thus, limitations of the available computing capacity partly prevent the spread of the LES method to the simulation of engineering applications with complex geometries and high Reynolds number flow. Therefore, hybrid turbulence models are proposed to combine the advantages of RANS and LES methods. In hybrid RANS/LES turbulence models, RANS and LES are used separately in regions where they are proper treatments for the local flow configuration.

Historically, the first hybrid methods have been introduced with the aim of tackling efficiently the problem of solving the wall region. The RANS method is considered as a convenient and economical treatment for regions near the wall. However, for the LES method, to explicitly compute the small-scale turbulence near solid boundaries, a mesh with a much smaller grid size than for RANS is needed. This makes LES computations extremely costly in terms of computer resources, especially for simulations with large domains or at large Reynolds numbers. On the other hand, the most challenging flow regions for RANS turbulence models are the regions where massive separation happens [43]. In these regions, the momentum-carrying turbulent eddies are not much smaller than the scale of the geometry, so that a turbulence-resolving flow description would not require very fine grids. Therefore, these regions are exactly where LES is most desirable. Thus, in hybrid RANS/LES methods, the idea is to treat the flow in the vicinity of walls by RANS and to use LES for the regions where the large-scale eddies dominate.

Since the time when Spalart et al. [90] presented the Detached-Eddy Simulation (DES) concept in 1997, extensive research effort has been dedicated to the development of hybrid RANS/LES methods. The proposed methods can be classified into two main categories in terms of the existence of an explicit RANS-LES interface between the two modeling types: zonal methods and non-zonal methods. In zonal methods which were proposed by Deck [23, 24], RANS is applied near solid walls to reduce the resolution requirement in the near-wall region, and LES is applied in regions far from the wall as well as in regions of separated flow. These regions are selected explicitly by the user and the interface is fixed and sharp. In zonal hybrid RANS/LES methods, different turbulence models can be employed in the two regions, e.g. a k-ω turbulence model in the RANS region and an algebraic subgrid scale model in the LES region. However, Spalart [91] considers that the choice of the RANS and LES zones is difficult for complex flows in the zonal approach, and that this approach should not be used for the simulation of too complex turbulent flows.

Unlike the zonal methods, for non-zonal methods the interface between RANS and LES regions is dynamic. In these methods, the turbulence model is switched dynamically from RANS to LES and vice versa based on a comparison between the local RANS and LES length scales. In general, these hybrid RANS-LES models simulate the flow using RANS if the local RANS length scale is smaller than the local LES length scale. Otherwise the flow is simulated using LES. DES97 [90] is considered as the first non-zonal hybrid RANS/LES turbulence model.

In addition, methods are proposed in which no interface exists to distinguish between RANS and LES modes, for example Partially-Averaged Navier-Stokes (PANS, see [38]) and Partially-Integrated Transport Method (PITM, see e.g. [17]). These methods are based on RANS models, which are modified to adopt aspects of a scale resolving model, i.e. LES-like behavior, when the turbulent fluctuations are strong and the grid resolution supports resolved turbulence. Otherwise the turbulent scales are modeled. Another example of turbulence-resolving modeling involving RANS and LES behavior without any explicit interfaces is the Scale-Adaptive Simulation model (SAS, see e.g. [61]), which should be seen as an URANS model with scale-resolving capability.

This thesis concentrates on non-zonal hybrid RANS/LES methods with an explicit RANS-LES interface, and especially on the DES turbulence model as a common and

relatively mature non-zonal hybrid RANS/LES method. In the following, a historical review of the evolution of this model is provided.

Detached Eddy Simulation (DES97). The original version of the DES model [90] is based on a modification to the Spalart-Allmaras (S-A) one-equation model [92]. The S-A model contains a destruction term for its eddy viscosity $\tilde{\nu}$ which is proportional to $(\tilde{\nu}/d)^2$, where d is the distance to the wall. In the DES formulation, the distance to the closest wall d is replaced by $\tilde{d}$ which is defined as:

$$\tilde{d} \equiv \min(d, C_{DES}\Delta), \tag{2.17}$$

where C_{DES} is a coefficient set to 0.65 which has been calibrated in decaying homogeneous turbulence [85] and Δ is the largest value of the local grid spacing in all three directions ($\Delta \equiv \max(\Delta x, \Delta y, \Delta z)$). Thus, a single turbulence model is defined that functions as S-A when $d \ll \Delta$, and when $\Delta \ll d$, the length scale is replaced by $C_{DES}\Delta$ so the model acts as a subgrid-scale (SGS) model. In many regions, especially the boundary layers, highly anisotropic grids are used. As a result, in boundary layers, although typically $\Delta y > d$, the local maximum grid spacing gives $d \ll \Delta$ which leads to RANS behavior. Conversely, in a grid which is adequate to resolve the eddies that arise after massive separation, the grid cells are more isotropic and $\Delta \ll d$ which produces LES behavior [90]. This model was tested in different aerodynamic flow cases such as a 2D mixing layer and the flow over a 2D backward facing step, the flow past an airfoil at different angles of attack up to 90° [85], and a circular cylinder including laminar or turbulent separation [99].

Although the original DES method is applied to the S-A model, which uses a distance to the wall as a turbulence length scale, its principle can be generalized to all RANS models by an appropriate adjustment in the reference length scale in order to modify the dissipation of the RANS turbulence model. As before, the basic idea relies on Eq. (2.17) where in this case, the turbulence length scale is computed as:

$$l_{DES} \equiv \min(l_{RANS}, C_{DES}\Delta), \tag{2.18}$$

where l_{RANS} denotes the length scale computed by the RANS mode. In the framework of DES using two-equation models, a formulation was proposed by Strelets [96] and Travin et al. [100] based on the k-ω SST RANS model [60]. The main reason for the selection of the k-ω SST model is its improved separation prediction capability with respect to S-A. The length scale of the k-ω SST model in terms of k and ω is:

$$l_{k-\omega} = k^{1/2}/(\beta * \omega), \tag{2.19}$$

where β is a k-ω turbulence constant. As for DES S-A, this length scale should be replaced with the DES length scale (Eq. 2.18). Then the basic two-equation k-ω SST model (Section 2.1.1) is transformed into the k-ω DES model by modifying the dissipation term in the kinetic energy equation (Eq. 2.10):

$$D^k_{RANS} = \rho\beta^* k\omega = \rho k^{3/2}/l_{k-\omega}, \tag{2.20}$$

replacing it by

$$D^k_{DES} = \rho k^{3/2}/l_{DES}. \tag{2.21}$$

This model was tested on the same cases as DES S-A, i.e. homogeneous decaying turbulence, airfoils beyond stall, a circular cylinder in subcritical and supercritical flow regimes, a backward facing step, and a model of a landing gear truck [96]. However, it did not show

any superiority over the original S-A based formulation [100]. Later, Menter et al. [63] presented a full formulation of the k-ω SST DES model by defining a blending function F_{DES} in the dissipation term of the k transport equation (Eq. 2.10) as:

$$D^k_{DES} = \rho\beta^* k\omega F_{DES}, \tag{2.22}$$

with

$$F_{DES} = \max(\frac{l_{k-\omega}}{C_{DES}\Delta}(1 - F_{SST}), 1), \tag{2.23}$$

where $l_{k-\omega}$ is the turbulent length scale given by Eq. (2.19). F_{SST} is a function selected from the blending functions of the SST model, F_1 (Eq. 2.13) and F_2 (Eq. 2.9). This function reduces the influence of the DES limiter (Eq. 2.23) on the boundary layer portion of the flow and for $F_{SST} = 0$ the Strelets SST DES model [96] is recovered. According to Eq. (2.22), whenever $F_{DES} = 1$ the model acts as RANS, and for $F_{DES} > 1$ the LES part is activated.

Delayed Detached Eddy Simulation (DDES). Over time and through the study of different test cases, the imperfections and limitations of the DES model were revealed. Since the proposal of the DES model, the RANS-to-LES interface where $d \approx C_{DES}\Delta$ and the modeling mode is changed from RANS to LES, has been studied. Across this interface, the modeling is switched from a non-turbulence resolving method to a turbulence-resolving method. The modeling mode is switched instantly across a single grid plane, the eddy viscosity is reduced, and therefore the modeled Reynolds stress drops. However, the resolved turbulent stresses associated with the resolved flow unsteadiness are not instantly fully developed to replace this reduction of the modeled turbulence. Since in the incoming RANS-simulated flow, there are no or only weak natural instabilities, a certain transition region downstream of the RANS-LES interface is needed for the LES-simulated flow to develop resolved turbulence - if it develops at all. This transition region was called *grey area* by Spalart et al. when they proposed DES [90]. The effect of this transition region was detected among the first test cases in which the DES model was used and is referred to as modeled-stress depletion (MSD). Squires et al. [95] reported the lack of eddy content in the detaching boundary layers, that represents a relatively small error in the solutions when a DES S-A model is used to predict the massively separated flow around a forebody cross-section. Menter et al. [63] discussed the significance of the quick development of the unsteady turbulent structures after the model has switched from the RANS to the LES mode for the prediction of separated shear layers.

The effects of the grey area become much more pronounced if the RANS mode is switched to LES inside the attached boundary layer. This occurs if the grid is gradually refined inside the boundary layer (e.g. when a user is seeking grid convergence), or when a boundary layer thickens and nears separation. Menter and Kuntz [62] showed that severe cases of MSD lead to grid-induced separation. They proposed a solution within the SST DES framework, using the two blending functions F_1 and F_2 to identify the boundary layer and prevent a switch to LES inside the boundary layer. Spalart [93] generalized the idea of Menter's proposal, which can be applied to any model as long as the eddy viscosity is involved. This modification is applied by redefining the DES length scale,

$$l_{DDES} = l_{RANS} - f_d \max\{0, (l_{RANS} - l_{LES})\}, \tag{2.24}$$

where l_{RANS} for the S-A model is equal to the distance to the wall d, and for the k-ω SST model it is given by Eq. (2.19). l_{LES} is defined proportional with the subgrid length

scale, $l_{LES} \propto \Delta$. The delaying function f_d based on F_1 and F_2 goes to zero in boundary layers, to ensure that DDES performs in its RANS mode in the major part of any attached boundary layer independent of the wall-parallel grid-spacing, thus eliminating the known incorrect DES performance on ambiguous grids (where $d \approx C_{DES}\Delta$). For f_d close to 1, the model reduces to the original DES97.

Improved Delayed Detached Eddy Simulation (IDDES). Another issue in turbulence simulation using DES comes from the accuracy that is required for the prediction of separation. With the basic DES premise that all turbulent boundary-layer physics is treated by RANS, this accuracy cannot surpass that of the RANS model. The DES equations, with fine enough LES grids in the boundary layer, can provide a wall modelling in LES (WMLES) and offer the possibility of breaking through this accuracy barrier of RANS at practical Reynolds numbers. In WMLES, RANS is used only in a thin near-wall region, in which the wall distance is much smaller than the boundary-layer thickness but is still potentially very large in wall units [73]. This motivated Nikitin et al. [72] to apply the raw DES equations to turbulent channel flow. However, they reported a buffer layer in the logarithmic velocity profiles around the vicinity of the RANS-LES interface. This means that the LES-predicted velocity profile does not match the RANS-predicted velocity profile due to the mismatch in the predicted turbulent stresses on each side of the RANS-LES interface. This issue, which is referred to as *log-layer mismatch*, is due to the rapid reduction of the turbulent viscosity across the RANS-LES interface and an underprediction of the resolved stresses on the LES side of the interface, as explained for DDES above. As a result of the log-layer mismatch, the skin friction is almost always underpredicted.

To solve this issue, Travin et al. [98] proposed a new version of the DES model, IDDES (Improved DDES), which includes two branches, DDES and WMLES, and a set of empirical functions designed to choose the correct branch given the circumstances. The DDES branch becomes active when the inflow into the LES zone does not have any turbulent content and in particular when a grid of boundary-layer type prevents the resolution of the dominant eddies. In this case, the length scale of the IDDES model should reduce to Eq. (2.24).

The WMLES branch of the IDDES model is intended to be active only when fine LES turbulent content is available at the inlet of the LES region and the local grid spacing is small enough to resolve the turbulent structures. This branch couples the RANS and LES approaches by introducing a blended RANS-LES length-scale,

$$l_{WMLES} = f_B(1 + f_e)l_{RANS} + (1 - f_B)l_{LES}, \tag{2.25}$$

where f_B and f_e are empirical blending functions [86] based on the cell size and the distance to the wall. f_B controls the activation of RANS and LES and provides rapid switching of the model from RANS mode ($f_B = 1$) to LES mode ($f_B = 0$) over the inner region of the boundary layer close to the wall. f_e is defined to prevent the excessive reduction of the RANS Reynolds stresses in the vicinity of the RANS and LES interface and to ensure that the log-layer mismatch does not occur.

The blending of the two branches DDES and WMLES is possible by defining $\hat{f}_d = \max\{(1 - f_d), f_B\}$ and formulating a modified version of the DDES length scale (Eq. 2.24), in a form similar to Eq. (2.25). Combined with Eq. (2.25), the IDDES length scale is implemented as

$$l_{IDDES} = \hat{f}_d(1 + f_e)l_{RANS} + (1 - \hat{f}_d)l_{LES}. \tag{2.26}$$

In a simulation with local inflow turbulent content, f_d is close to 1.0, so $\hat{f}_d$ is equal to f_B and the IDDES length scale reduces to Eq. 2.25 ($l_{IDDES} = l_{WMLES}$). Otherwise, the equivalent of Eq. (2.24) is obtained.

2.3 Adaptive Grid Refinement

Adaptive grid refinement (AGR), refers to the automatic modification of an existing mesh during a computation, to accurately capture flow features. It is an ideal way to efficiently solve flow problems that have strong local structures whose position is not known *a priori*, without an excessive increase in computational effort. Otherwise, to treat a problem with these structures, a uniform mesh with very fine grids for the entire domain must be generated to resolve accurately all the available structures. However, there might be areas where the flow changes little and could be resolved accurately enough on a much coarser grid. The adaptive grid refinement helps to have a fine grid only in areas where it is needed and to calculate the remainder of the solution on a coarser grid. This means that the entire solution is resolved accurately, but no unnecessarily accurate calculations are made for the areas where the flow is more or less constant.

The adaptation process is performed in a variety of ways, but these all have three main ingredients: (1) the refinement strategy, which corresponds to the method of applying the refinement or coarsening, (2) the refinement criterion, which determines the cells or elements that are to be treated by adaptation, and (3) the refinement threshold, a global parameter which indicates how much refinement/coarsening is required.

Refinement strategy. Adaptation methods which alter the mesh to follow the flow features fall under two broad classifications,

- relocating or moving a mesh (r-refinement),
- mesh enrichment:
  - changing locally the order of accuracy (p-refinement),
  - local refinement and/or coarsening of a mesh (h-refinement).

Mesh movement, or r-refinement, maintains the mesh topology but allows mesh lines to move inside the domain. Thus, as the mesh moves, some cells are contracted while simultaneously others are expanded [59]. The advantage of this technique is that the number of mesh cells and therefore the computational cost does not increase when a new flow field is calculated on the adapted mesh. In addition, for parallel computations, the mesh decomposition only needs to be carried out once. On the other hand, this method is generally not capable of finding a solution with a specified accuracy [27, 34]. If the mesh is too coarse, it might be impossible to achieve a high degree of precision without more refinement. In addition, the method is not flexible and general enough for simulations with complex geometries.

The second method of mesh adaptation is called mesh enrichment. In this method, degrees of freedom are added or taken from a mesh. One may either split the cells in the areas where refinement is needed, which is referred to as h-refinement, or add further degrees of freedom with hierarchical shape functions (p-refinement). The p-refinement method is generally used with the discontinuous Galerkin (DG) method rather than the finite-volume (FV) method [51], since it requires by definition a high-order discretisation where the number of degrees of freedom can be easily varied. Efficient p-refinement can reduce dissipation and dispersion errors in regions where the solution is smooth. Thus, the accurate resolution of turbulence can be obtained with few degrees of freedom [71]. This method can also be employed in combination with h-refinement in regions where geometrical and physical discontinuities occur [18, 22].

Early developments of adaptive refinement dealt almost exclusively with h-refinement on structured grids [11, 12]. This refinement type is by far the most successful mesh enrichment strategy and has been explored by numerous investigators and applied in a variety of different contexts, see for example [3, 28, 109].

The way to perform h-refinement is highly dependent on the type of the mesh. For quadrangles and hexahedra, the cells can simply be cut through the middle to produce smaller cells. For meshes of triangles or tetrahedra however, the refinement and the place of new vertices need to be applied carefully according to specific constrains to preserve the quality and size of the triangles, and element modifications such as edge swaps may be required (e.g. [2, 6]). For most unstructured tetrahedral meshes, when the grid is refined the cells get more and more the same shape and size as their neighbors. In unstructured hexahedral mesh refinement however, small cells are always twice finer than their larger neighbours and hanging vertices (Figure 2.1) are inevitable. Even though this may reduce the quality of the state reconstruction at the faces [78, 106], such topologies are inherent in hexahedral refinement.

Finally, h-refinement can be applied as isotropic or directional refinement. For isotropic refinement, cells are refined in all their directions at once (a quadrangle is divided in four, a hexahedron in eight, etc.), the resulting refined small cells have the same shape as the original cells. In anisotropic refinement, the directions of the divisions or the aspect ratios of the cells can be controlled as well. Unlike isotropic refinement which is often expensive in three dimensions, especially when discontinuities need to be captured, anisotropic meshes are effective due to the significant reduction in the number of degrees of freedom provided by oriented meshes. Numerous works in the literature explore the anisotropic type of h-refinement [7, 52, 56], representing the anisotropic parameters by defining metric tensors [13, 16, 36] to perform directional refinement.

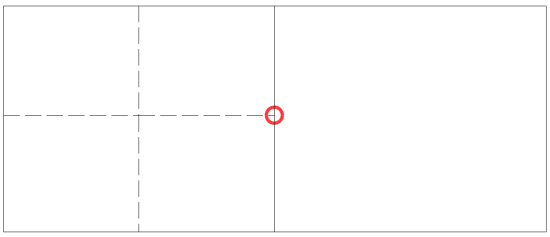


Figure 2.1 – The hanging vertex in a 2D example

Refinement criterion. Besides the refinement strategy, any adaptive refinement scheme contains another main ingredient: an error indicator/estimator, also referred to as the refinement criterion. The first stage in a simulation with adaptive refinement consists in creating an initial mesh of a given domain, which is used to perform an initial computation. Then, the cells or elements that require refinement or coarsening should be identified. The refinement criterion decides which parts of the grids are to be refined or de-refined. Error indicators/estimators can be divided into several categories with respect to what they are based on:

- Feature-based criteria,
- Truncation error-based criteria,
- Adjoint-based criteria,
- Hessian-based criteria.

Featured-based indicators are developed based on gradients or curvature information of the flow variables such as the pressure and the velocity, or derived quantities such as vortex tracking criteria [66] and free surface criteria [107]. They are straightforward to implement since the indicators can be computed directly from the evolving flow field, which is readily available. However, continuous local refinement of the dominant flow features does not necessarily guarantee the simultaneous reduction of the global error. In certain cases this procedure may even lead to incorrect results (e.g. [32]).

Truncation error can also be used as a driver for the adaptation process. With this error estimator, the aim is to minimize the difference between the partial differential equation and its discrete approximation. The general concept behind truncation error-based adaption is to equi-distribute the truncation error over the entire domain to reduce the total discretization error [46, 47]. However, it is difficult to evaluate a truncation error accurately since it requires a higher-order discretization. In addition, truncation error-based criteria are inherently isotropic criteria.

Another promising method for grid adaptation is the adjoint approach. Adjoint methods estimate the local contribution of each cell to the discretization error in any solution functional of interest (e.g., lift, drag, and moments). Thus, it can provide targeted mesh adaption depending on the goals of the simulation [102]. This type of criteria is based on a weighting of other criteria types like truncation error estimation or Hessian-based criteria [6, 57]. The main drawback for adjoint methods is their complexity.

Hessian-based criteria can be considered as a separate class of error indicators. These criteria are based on the interpolation error estimate for linear reconstructions, which means that they give an indication of the truncation errors. This anisotropic metric is a function of the Hessian of the solution which is reconstructed from the numerical solution (second-order derivatives of the numerical solution) [36]. In adaptation based on Hessian criteria, the usual objective is to minimize the maximum of the interpolation errors or to minimize the interpolation errors for some solution fields [2, 35, 55]. As noted above, Hessian indicators are often used in an adjoint-based framework.

Threshold. Finally, most refinement methods contain a parameter which globally determines how fine the mesh will be: the refinement threshold. In practice, to reach a good balance between the accuracy and the computational time, the refinement threshold should be well-chosen. If this parameter is set too small, the results will be accurate but the computational cost will be expensive. On the contrary, if this threshold is set too large, pertinent regions are not detected and the mesh adaptation algorithm will not be efficient. Usually, the mesh refinement threshold is a test-case dependent parameter. Several research works were carried out to avoid hand-calibration of the refinement threshold. For instance, Pons and Ersoy [75] proposed an automatic thresholding based on a distribution function to set the threshold parameter regardless of the type of refinement criterion. Alauzet and Loseille [2] introduced a complexity parameter, instead of the threshold, as a measure to determine the number of cells after refinement.

2.3.1 Coupling AGR and DES

In simulation using DES-type models, the generation of a mesh with an appropriate resolution is very important since a too coarse mesh does not resolve the flow, while a too fine mesh may lead to modeled stress depletion. Although the improvement in these turbulence models reduces the sensitivity of these models to the local resolution of the mesh, it is not easy for a user to choose the properties of a mesh without knowing the flow behavior. This turns to a complex puzzle if the adaptation process is coupled with DES-type simulations. On one hand, the mesh changes frequently as AGR adapts the

local grids with respect to the flow features. These changes in the mesh can perturb the turbulence model and pollute the solution. On the other hand, AGR can help ensuring that the right mesh size for a DES model is produced everywhere.

The combination of adaptive refinement and RANS turbulence models is fully developed and has been used broadly [64, 112, 115]. In addition, despite the discussion on the reliability of the combination of AGR and LES, due to the discontinuities which happen for resolved solutions at grid refinement interfaces between coarse and fine grids [14, 40, 74, 101], these two techniques have been successfully used together especially for relatively complex geometries where an adequate mesh has to be determined to limit the computational cost [9, 97]. The main research topic today is to select the right refinement criterion to ensure a correct resolution of the whole resolved field for the LES simulation. Daviller et al. [21] defined a metric based on the time-averaged value of the kinetic energy dissipation which determines the local mesh size for h-refinement. Toosi and Larsson [97] derive an anisotropic Hessian-like criterion by considering the equidistribution of the small-scale turbulent energy for each cell over the cell directions. In the context of p-refinement, an analysis of different refinement indicators for discontinuous Galerkin methods was carried out by Naddei et al. [71].

In this thesis, the objective is to assess the possibility of coupling AGR and DES-type turbulence models as they were defined in the ISIS-CFD solver. Like in LES simulations, to ensure the reliability of the solution, the assessment of the grid resolution in hybrid RANS/LES methods is crucial especially for LES regions that may not be known in advance. Despite a few recent contributions on using adaptive refinement for hybrid RANS/LES, like the proposal by Reuß et al. [80] of a resolution sensor instead of the original refinement criterion in the adaptation process to detect the LES regions, in the most cases a stationary mesh with locally find grids are used with DES-type turbulence models [49, 113].

Chapter 3

The ISIS-CFD flow solver

The development in this thesis is based on the ISIS-CFD flow solver. ISIS-CFD is an incompressible unsteady multifluid Navier-Stokes solver based on the finite volume method, which was developed by the METHRIC team (Modélisation des Écoulements Turbulents à Haut Reynolds Incompressibles et Couplages) in the LHEEA lab of Centrale Nantes and CNRS and is distributed by NUMECA Int. as a part of the FINE™/Marine computing suite, which is mainly devoted to marine hydrodynamics. Free-surface flows are simulated by incompressible and non-miscible flow phases, modeled through the use of conservation equations for the volume fraction of each phase.

For the Reynolds-averaged Navier-Stokes equations (RANS), the method features the classical two-equation k-ω SST model (Section 2.1.1), as well as the anisotropic two-equation Explicit Algebraic Reynolds Stress Model (EARSM) [31] and Reynolds Stress Transport Models [26], with or without rotation corrections. All models are available with wall-function or low-Reynolds near wall formulations. hybrid RANS/LES turbulence models based on Detached Eddy Simulation (DES) (Section 2.2) are also implemented and have been validated on automotive flows with large separations (see [43]).

An adaptive mesh refinement procedure is integrated in ISIS-CFD. Hexahedral original meshes used by ISIS-CFD can be modified dynamically during computations. The adaptation method uses anisotropic h-refinement based on Hessian-based criteria (Section 2.3) for refinement decisions. For multifluid simulations, free-surface capturing mesh refinement is also available in the solver.

This chapter begins with a short description of the governing equations in Section 3.1. The discretization schemes for these equations are discussed in Section 3.2 along with the algorithm for solving the discretized system. Finally, in Section 3.3, the adaptation process in ISIS-CFD is presented.

3.1 Governing equations

This section presents the governing equations of the ISIS-CFD solver, which allow the flow solver to deal with two-phase flows and moving grids. In the two-phase continuum for incompressible flow of viscous fluid under isothermal conditions, using the generalized form of Gauss' theorem, the RANS equations (Eq. 2.5) and the conservation of phase 1 can be written as:

$$\frac{\partial}{\partial t}\int_V \rho U_i dV + \int_S \rho U_i (U_j - U_{d_j}) \cdot n_j dS = \int_S (\tau_{ij} - P\delta_{ij}) \cdot n_j dS + \int_V \rho g_i dV, \tag{3.1a}$$

$$\frac{\partial}{\partial t}\int_V \rho dV + \int_S \rho (U_i - U_{d_i}) \cdot n_i dS = 0, \tag{3.1b}$$

$$\frac{\partial}{\partial t}\int_V \alpha dV + \int_S \alpha(U_i - U_{d_i}) \cdot n_i dS = 0, \tag{3.1c}$$

where V is the domain of interest, or control volume, bounded by the closed surface S moving at the velocity U_{d_i} with a unit normal vector n_i directed outward. U_i and P represent, respectively, the averaged velocity and pressure fields. τ_{ij} and g_i are the components of the (viscous and turbulent) stress tensor and the gravity vector, whereas δ_{ij} is a delta function which is equal to unity only for $i = j$. α is the volume fraction for the fluid 1 and is used to distinguish the presence ($\alpha = 1$) or the absence ($\alpha = 0$) of fluid 1. Since a volume fraction between 0 and 1 indicates a mixture of two fluids, the value of 0.5 is selected as a definition of the interface between these two fluids.

The effective flow physical properties (viscosity and density) are obtained from the properties of each phase (μ_i and ρ_i) with the following constitutive relations:

$$\rho = \alpha\rho_1 + (1-\alpha)\rho_2, \quad \mu = \mu\rho_1 + (1-\mu)\rho_2. \tag{3.2}$$

When the grid is moving, the so-called *space conservation law* must also be satisfied:

$$\frac{\partial}{\partial t}\int_V dV - \int_S U_{d_i} \cdot n_i dS = 0. \tag{3.3}$$

A simplified form of the general mass conservation equation (3.1b) can be obtained when considering incompressible phases with constant densities ρ_i [78]:

$$\int_S U_i \cdot n_i dS = 0, \tag{3.4}$$

which is the equivalent of Eq. (2.5b). Thus, even the multi-phase fluid has a divergence-free velocity field.

The definition of the turbulent Reynolds stresses for the closure of the Reynolds-averaged equations, as well as the k-ω SST turbulence model, are discussed in Section 2.1. In addition, the DES-based models available in ISIS-CFD are DES-SST, DDES and IDDES. These models are presented in Section 2.2.

3.2 Numerical framework

This section shows the basic approach used to transform the flow equations of the preceding section into a discretized system of equations for the velocity, pressure, and volume fraction. The unstructured discretization is face-based and the method is generalized to two-dimensional, rotationally-symmetric, or three-dimensional unstructured meshes for which non-overlapping control volumes are bounded by an arbitrary number of constitutive faces. The velocity field is obtained from the momentum conservation equations and the pressure field is extracted from the continuity equation, transformed into a pressure-equation. In the case of turbulent flows, the additional transport equations for the modeled variables are discretized and solved using the same principles.

The pressure equation is derived from the conservation equation (3.4) where a coupling between the pressure gradient and the velocity at the faces [78] is introduced in the volumetric flux reconstruction $U_i \cdot n_i dS$. While the methodology is based on the Rhie and Chow SIMPLE [81] (Semi-Implicit Method for Pressure-Linked Equations) algorithm, special attention is given (i) to the pressure equation formulation in presence of a density discontinuity and (ii) the way unsteady terms are interpolated so that, when an overall

steady solution is expected, the solution does not depend on the time step Δt. The originality of the method proposed by [78] is to take into account the pressure gradient discontinuity by using a pressure equation based on $\nabla p/\rho$ rather than ∇p itself in the context of the SIMPLE algorithm.

3.2.1 Discretized form of the momentum equations

For a cell V with center C which is limited by an arbitrary number of faces f shared with neighbor cells nb (Figure 3.1), the momentum equation (3.1a) in vector notation can be written in the form:

$$\frac{\delta}{\delta\tau}(\rho V\mathbf{U})_C + \frac{\delta}{\delta t}(\rho V\mathbf{U})_C + \sum_f (F_{c_f} - F_{d_f}) = S_Q^V. \tag{3.5}$$

The terms F_{c_f} and F_{d_f} are respectively the convective and diffusive fluxes across the face f. S_Q^V is the volume source term.

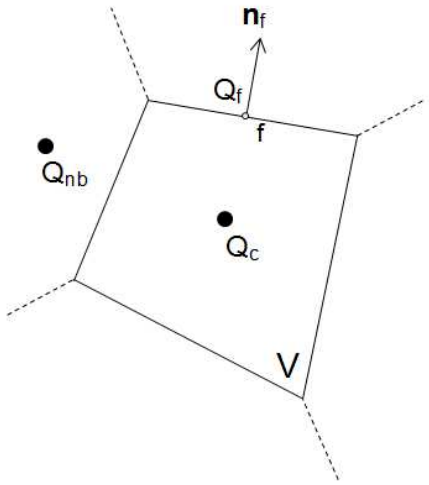


Figure 3.1 – Control volume notations, Q stands for generic quantities at the cell centers Q_c and Q_{nb}, or at the surface of a cell Q_f

For the implicit time integration, the temporal derivatives based on the Adams–Bashforth second-order upwind temporal scheme [30] are evaluated by:

$$\frac{\delta}{\delta t}(\rho V\mathbf{U})_C \cong e^c(\rho V\mathbf{U})_C^c + e^p(\rho V\mathbf{U})_C^p + e^q(\rho V\mathbf{U})_C^q, \tag{3.6}$$

where the subscripts c, p and q stand for the current and the two previous time steps in the time scheme. The weights of these time steps, e^c, e^p and e^q, are obtained from the Taylor series expansion based on the current time as $3/2\Delta t$, $-4/2\Delta t$ and $1/2\Delta t$ respectively. The first term of the left-hand side of equation (3.5) is a pseudo-unsteady term needed to stabilize the solution procedure for steady flows. The corresponding derivative is evaluated by:

$$\frac{\delta}{\delta\tau}(\rho V\mathbf{U})_C = \frac{(\rho V\mathbf{U})_C^c - (\rho V\mathbf{U})_C^{c0}}{\Delta\tau}, \tag{3.7}$$

where $(\rho V\mathbf{U})_C^{c0}$ is the previous estimation of $(\rho V\mathbf{U})_C^c$ within the non-linear loop.

The convective fluxes are expressed as the product of the volumetric flux (see Section 3.2.2) and a special reconstruction of the cell-centered values to the faces. To guarantee

accuracy, stability and boundedness of the solutions, these schemes are developed in the Normalized Variable Diagram (NVD) [54]. The NVD diagram is based on points U, C and D which represent the downwind, central, and upwind cell centers in the neighborhoods of a face (Figure 3.2). The idea of the NVD is to represent a reconstruction scheme for the face value as a function of $\tilde{Q}_C$, the normalized value in the central cell:

$$\tilde{Q}_C = \frac{Q_C - Q_U}{Q_D - Q_U}. \tag{3.8}$$

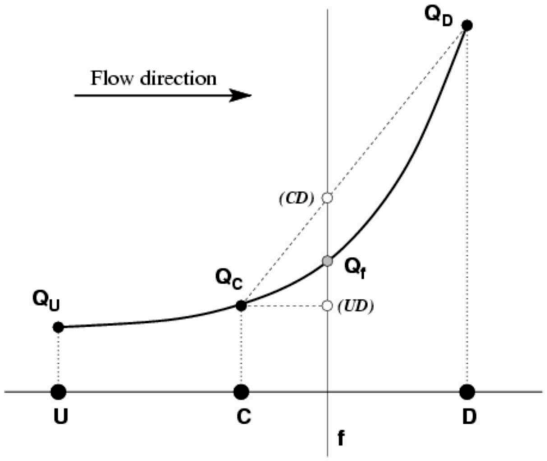


Figure 3.2 – 1D variation of a convected quantity Q in physical space

On arbitrary unstructured grids, the far upstream node U is not known explicitly (C and D are the centers of the two cells next to the face, see Figure 3.2). Therefore, an alternative is chosen: an imaginary nodal quantity Q_U is defined by the use of the gradient projection method [78] in such a way that:

$$Q_U = Q_C - \overrightarrow{CU} \cdot \nabla Q|_C, \quad \text{with} \quad \overrightarrow{CU} \triangleq -\overrightarrow{CD}. \tag{3.9}$$

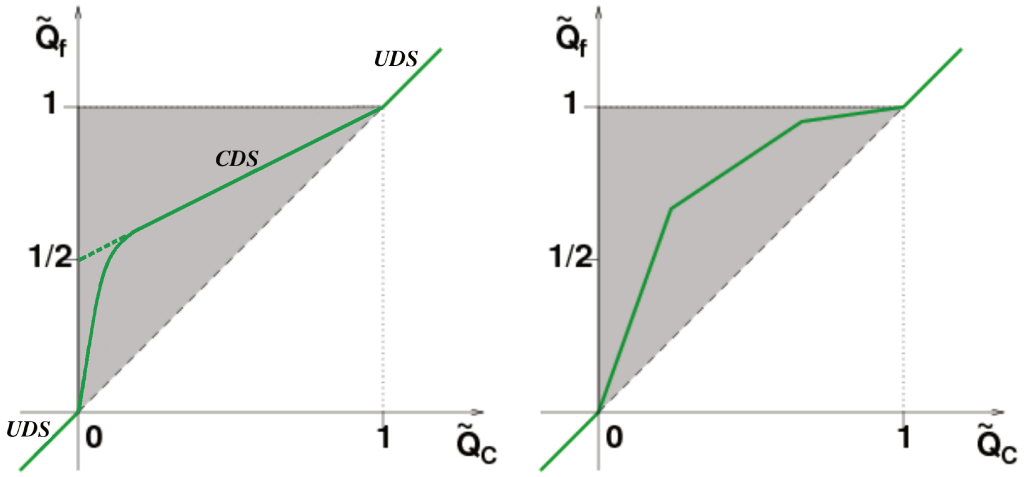
With the newly defined upstream quantity, $\tilde{Q}_C$ can be re-evaluated as:

$$\tilde{Q}_C = 1 - \frac{Q_D - Q_C}{2\nabla Q|_C \cdot \overrightarrow{CD}}. \tag{3.10}$$

Now the non-dimensionalized quantities at the face $\tilde{Q}_f$ are defined as a function of $\tilde{Q}_C$ based on available schemes. In ISIS-CFD, the discretization of the convective fluxes in both the momentum equations and the equations for turbulence modelling is usually based on one of two implemented schemes: (1) the Gamma Differencing Scheme (GDS) [50] (Figure 3.3a) which is a blending of the first-order upwind differencing scheme (UDS) and the second-order centered differencing scheme (CDS), (2) the AVLSMART scheme [77] (Figure 3.3b).

After replacing the discretized terms in Eq. (3.5) and isolating the pressure gradient together with the gravity acceleration, the discretized momentum equation for any Cartesian velocity component reads:

$$\begin{aligned} &(e^c + \frac{1}{\Delta\tau_C})(V\rho\mathbf{U})^c_C + (eV\rho\mathbf{U})^p_C + (eV\rho\mathbf{U})^q_C \\ &+ a_C\mathbf{U}^c_C + \sum_{nb} a_{nb}\mathbf{U}^c_{nb} + \mathbf{S}_C + (V\nabla p)^c_C \\ &= (\rho V)^c_C\mathbf{g} + \frac{(\rho V)^c_C\mathbf{U}^{c0}_C}{\Delta\tau_C}, \end{aligned} \tag{3.11}$$



(a) Gamma differencing scheme (b) AVLSMART scheme

Figure 3.3 – Discretization schemes in NVD diagram

where a_C and a_{nb} are the matrix coefficients from the implicit part of the diffusive and convective terms. $\mathbf{S}$ is a source term containing all explicit remaining contributions and external force fields except gravity and pressure.

3.2.2 Fluxes and pressure equation

To derive volumetric fluxes, first the momentum equations are rewritten to express the velocity at the cell-center:

$$\begin{aligned} \mathbf{U}_C^c = & -Cp_C \left(\hat{\mathbf{U}}_C + (\nabla p/\rho)_C^c - \mathbf{g}\right) + Cp_C \mathbf{U}_C^{c0}/\Delta\tau_C \\ & -Cp_C \left((eV\rho\mathbf{U})_C^p + (eV\rho\mathbf{U})_C^q\right)/(\rho V)_C^c, \\ \text{with} \quad Cp_C = & \left(e^c + \frac{1}{\Delta\tau_C} + \frac{a_C}{(\rho V)_C^c}\right)^{-1}, \end{aligned} \tag{3.12}$$

where the discretized vector $\hat{\mathbf{U}}$, homogeneous to gravity acceleration, includes part of the diffusion, convection and source terms. Unsteady and pseudo-unsteady contributions are explicitly kept and not transfered into this intermediate vector.

$$\hat{\mathbf{U}}_C = \left(\sum_{nb} a_{nb}\mathbf{U}_{nb}^c + \mathbf{S}_C\right)/(\rho V)_C^c. \tag{3.13}$$

Reconstruction of volumetric fluxes. The velocity vector on the cell faces is expressed as the face interpolate of equation (3.12):

$$\begin{aligned} \mathbf{U}_f^c = & -Cp_f \left(\hat{\mathbf{U}}_f + (\nabla p/\rho)_f^c - \mathbf{g}\right) + Cp_f \mathbf{U}_f^{c_0}/\Delta\tau_f \\ & -Cp_f \left(e^p(\rho V)_f^p \mathbf{U}_f^p + e^q(\rho V)_f^q \mathbf{U}_f^q\right)/(\rho V)_f^c, \\ \text{with} \quad Cp_f = & \left(e^c + \frac{1}{\Delta\tau_f} + \frac{(a_C)_f}{(\rho V)_f^c}\right)^{-1}. \end{aligned} \tag{3.14}$$

Except the pressure gradient, all terms in the previous equation are interpolated from the available cell quantities (L, R) on both sides of the face with a central operator $\mathcal{CLR}()$. For instance,

$$Cp_f = \left(e^c + 1/\mathcal{CLR}(\Delta\tau) + \mathcal{CLR}(a_C)/\mathcal{CLR}(\rho V)\right)^{-1}.$$

Then, the reconstruction of volumetric fluxes $\mathcal{F}(\overrightarrow{U})$ reads:

$$\mathcal{F}(\mathbf{U}) = S_f \mathbf{U}_f^c \cdot \mathbf{n}_f. \tag{3.15}$$

Pressure equation. When the fluxes (3.15) are substituted into the continuity equation (3.4), the discretized pressure equation is obtained as:

$$\begin{aligned} -\sum_f C_{p_f} \nabla P|_f S_f \mathbf{n} = & \sum_f C_{p_f} \left(\hat{\mathbf{U}}\right) S_f \mathbf{n} \\ & - \sum_f C_{p_f} \left(\frac{(\rho \mathbf{U})_f^{c_0}}{\Delta \tau_f} \right) S_f \mathbf{n} \\ & + \sum_f C_{p_f} \left(\frac{(e\rho V \mathbf{U})_f^p + (e\rho V \mathbf{U})_f^q}{V_f^c} \right) S_f \mathbf{n}, \end{aligned} \tag{3.16}$$

where the flux of the pressure gradient across the face f is obtained by a central discretization plus non-orthogonal corrections [78]. The matrix assembled from all control volumes is sparse, symmetric, and positive definite so that conjugate gradient based iterative solvers can be used.

3.2.3 Algorithm

The discretization of mass and momentum conservation equations yields a set of algebraic equations: one for each control volume and for each transport/conservation equation. These nonlinear and coupled equations are solved by the following segregated algorithm (Figure 3.4):

1. Initialize flow field quantities Q^0 at $t = t^0$.
2. New time step $t = t + \Delta t$,
3. If needed for this time step, call the mesh adaptation,
4. Start the iterative procedure with $Q = Q^0$,
5. If needed, compute the volume fraction for each fluid phase and update the global fluid properties from Eq. (3.2),
6. If needed, compute the turbulent quantities from the field of step 4,
7. Solve the momentum equations (3.11) to obtain a new prediction of the velocities,
8. Solve the pressure equation (3.16) to obtain a new pressure field,
9. Update the velocity face fluxes (3.15) and correct the velocity components (3.12) with the new pressure field,
10. If the nonlinear residuals are not low enough, go to step 4 and update the iteration counter within the time step,
11. Go to step 2 and update the time, t.

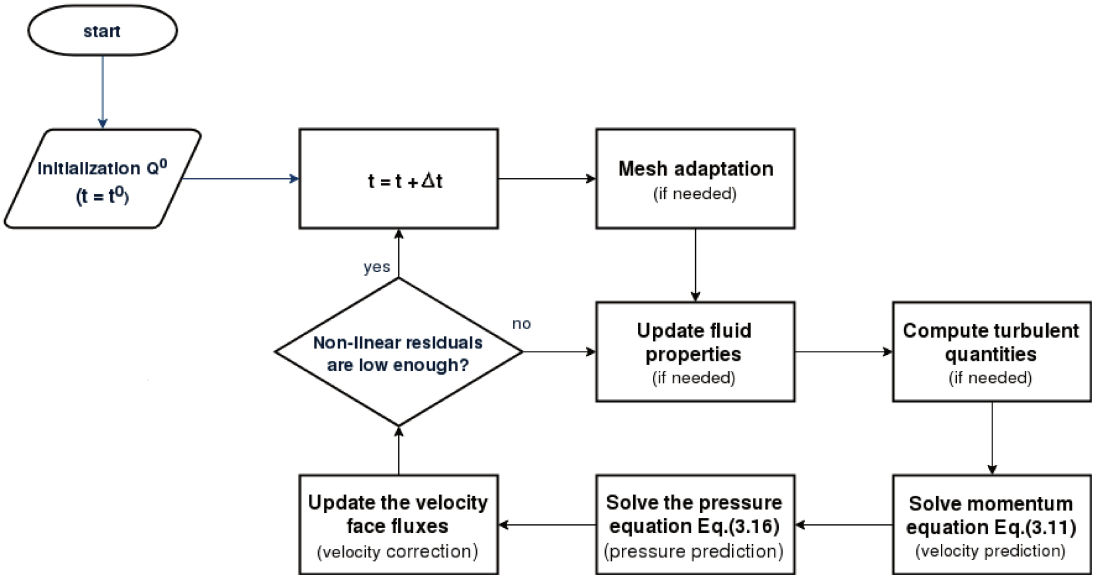


Figure 3.4 – Flow chart of the ISIS-CFD solver algorithm

Concerning the linear solver used in steps 5, 6, 7 for the phase concentration, turbulent quantities, and velocity components respectively, about 20 Gauss-Seidel iterations are enough to converge when the diagonal dominance is increased by 50% with the help of the pseudo-unsteady term.

While this pointwise solver works well for linear systems arising from discretized transport equations, its efficiency is dramatically reduced for the pressure operator. This operator is elliptic in nature and requires the complete solution of a Laplace-like equation for each step. Moreover, the corresponding linear system is ill-conditioned for highly stretched grids. This is why the flow solver uses either an Algebraic Multigrid solver or an efficient PGMRES algorithm with a Krylov subspace of dimension 5 and an Incomplete LU(k) preconditioning for the pressure equation.

3.3 Adaptive grid refinement in ISIS-CFD

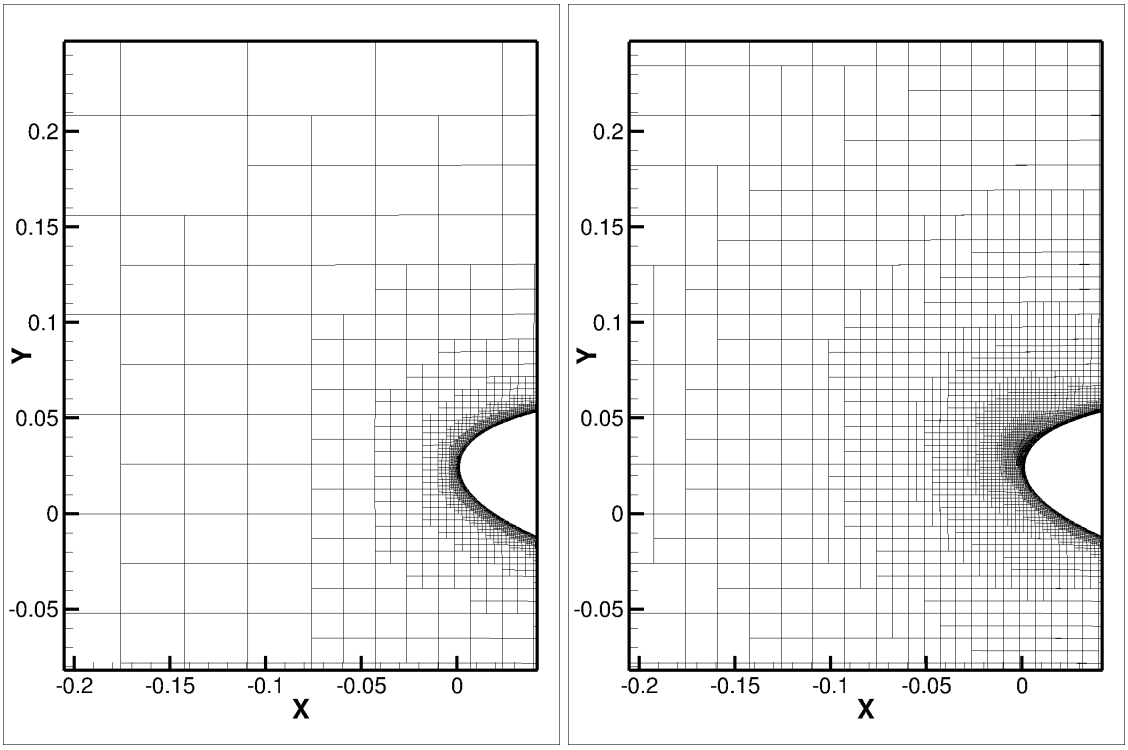
The grid refinement procedure developed for ISIS-CFD is integrated completely in the flow solver. The method is entirely parallelized, including automatic redistribution of the grid over the processors.

3.3.1 Refinement type

ISIS-CFD is a second-order finite volume code with complex geometries. Thus, among the available refinement methods described in the previous chapter (Section 2.3), h-refinement is the best choice. Relocating a mesh (r-refinement) for arbitrary complex geometries is difficult and p-refinement requires to increase the order of accuracy, which is not possible with the current version of the finite-volume discretization. On the contrary, locally refining or coarsening grids is a method which has good compatibility with ISIS-CFD without any need to modify the solver.

Original meshes used in ISIS-CFD are generated with HEXPRESS™ which is an unstructured full-hexahedral meshing tool developed by NUMECA International. This leads naturally to adaptive refinement based on hexahedral division. The first advantage

of this refinement is that the final adapted meshes look just like an original HEXPRESS-generated mesh (Figure 3.5). Thus, they can be used directly in the flow solver without further modifications by meshing tools.



(a) Original HEXPRESS mesh (b) Adapted mesh

Figure 3.5 – The similarity between an original HEXPRESS-generated mesh and a refined mesh after adaptation

The second advantage is, that refinement by hexahedral division is computationally cheap. Therefore, during a flow computation with ISIS-CFD, the refinement procedure can be called repeatedly. In such a call, first the refinement criterion is calculated based on the current flow solution, to indicate which parts of the grid are not fine enough and require refinement, or are too fine so that the grids can be derefined. Then in a separate step of the procedure the grid is (de)refined based on this criterion and the solution is copied to the new grids. This cycle is repeated several times to further refine or to derefine the mesh (i.e. to undo earlier refinements). Thanks to the repeated calls, the flow and the mesh converge together.

Grid refinement in ISIS-CFD is anisotropic. Isotropic refinement is very costly in three dimensions: for instance, for a hexahedron each refinement means a division in eight. Thus, creating very fine cells to accurately resolve a local flow phenomenon becomes difficult. However, by applying anisotropic refinement for flow features that need a fine grid in only one direction (e.g. the water surface), the total number of cells required can be greatly reduced or much finer flow details can be resolved.

3.3.2 Refinement criterion

The computation of the refinement criterion is decoupled from the decision of which cells to refine. Therefore, it is easy to exchange refinement criteria without modifying the remainder of the method. In order to offer a flexible framework for the specification of anisotropic grid refinement, a metric-based method of criterion evaluation [36, 37] was adopted in ISIS-CFD,

in which the refinement criteria are specified as tensors. For tensor-based refinement, the refinement criterion in each cell is a 3×3 symmetric positive definite matrix C_i which is considered as a geometric operator. This operator transforms each cell Ω_i in the physical space into a deformed cell $\tilde{\Omega}_i$ in a modified space (Figure 3.6). In each hexahedral cell, the cell size vectors $\mathbf{d}_{i,j}$ $(j = 1, ..., 3)$, which are the vectors between the opposing face centers in the three cell directions, are modified by the refinement tensor as:

$$\tilde{\mathbf{d}}_{i,j} = C_i \mathbf{d}_{i,j}, \tag{3.17}$$

where $\tilde{\mathbf{d}}_{j,i}$ are the modified sizes of the cell. A cell i is refined in the direction j when the modified size exceeds a given, constant threshold value T_r:

$$\|\tilde{\mathbf{d}}_{i,j}\| \geq T_r. \tag{3.18}$$

In opposite, a previously refined group of cells can be derefined in the direction j if $\tilde{\mathbf{d}}_{j,i}$ is lower than T_r/d for all cells in the group. The constant d is chosen slightly larger than 2, to prevent cells being alternately derefined and re-refined.

Several refinement criteria are available in ISIS-CFD, for example a criterion that refines at the water surface and criteria based on Hessian matrices.

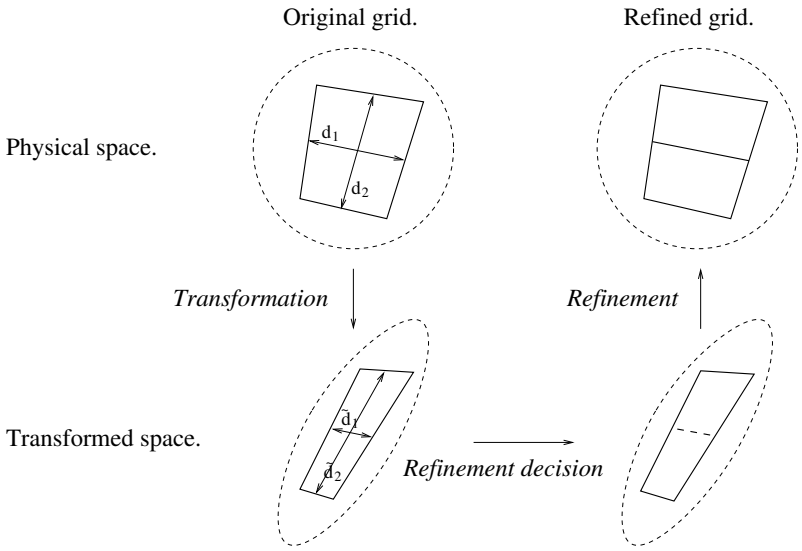


Figure 3.6 – Tensor refinement criterion. Cell Ω_i and unit circle (reference) in the physical space, deformed cell $\tilde{\Omega}_i$ and deformed circle after application of the transformation C, refinement decisions to create a uniform grid in the deformed space, and the resulting anisotropically refined grid

Free surface criterion. To resolve accurately the solution of Eq. (3.1c) which is a discontinuity for α that is convected with the flow, it is sufficient to refine the grid at and around the free surface, in the direction normal to the surface. Where the free surface is diagonal to the grid directions, isotropic refinement is used, but where the surface is horizontal, directional refinement is chosen [109] (Figure 3.7). The free-surface criterion C_S is based on α in the cells, it is non-zero when α is neither 0 nor 1. The normal direction to the surface is computed from a field α_s which corresponds to α, smoothed out by averaging over a cell and its neighbors a given number of times. The gradient of this field gives the normal directions. The criterion is then derived from vectors $\mathbf{v}_\alpha$ in each cell which are unit

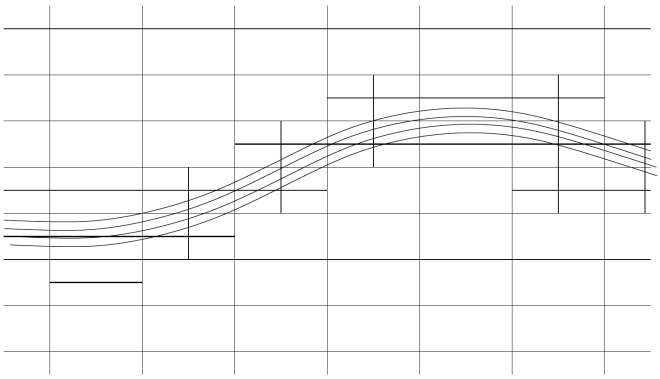


Figure 3.7 – Isotropic and directional refinement at the free surface

vectors in this normal direction for those cells where the smoothed α_s field is non-zero:

$$\mathbf{v}_\alpha = \frac{\nabla \alpha_s}{\|\nabla \alpha_s\|}. \tag{3.19}$$

Using the smoothed field guarantees that the normals are well-defined and also that the mesh is refined in a certain zone around the surface to create a margin of safety. Where the free-surface criterion is non-zero, it is computed as matrices having only one positive eigenvalue, associated with the direction of the vector $\mathbf{v}_\alpha$. The tensors C_S are computed as follows (with $\otimes$ representing the tensor product):

$$C_s = \mathbf{v}_\alpha \otimes \mathbf{v}_\alpha. \tag{3.20}$$

Pressure Hessian criterion (PH). Hessian matrices can be interpreted as rough error indicators, since they are linked to interpolation errors for linear interpolation [56]. Thus, they provide a measure of the truncation error for a second-order finite-volume discretization. The Hessian matrix $H(q)$, for a given state variable q, is a 3×3 symmetric tensor:

$$H(q) = \begin{bmatrix} q_{xx} & q_{xy} & q_{xz} \\ q_{xy} & q_{yy} & q_{yz} \\ q_{xz} & q_{yz} & q_{zz} \end{bmatrix}. \tag{3.21}$$

The Hessian matrix can be used directly as a tensor refinement criterion. In ISIS-CFD however, assuming that an indication of the local error is given by $H(q)$ times the cell sizes squared (which is reasonable for a second-order accurate discretization), equidistribution of this error indicator leads to [109]:

$$C_{PH} = \left(\|H(p)\|\right)^n, \tag{3.22}$$

where $n = 1/2$. The absolute value of a matrix $\|\cdot\|$ corresponds to a matrix having the same eigenvectors as the original one and the absolute values of its eigenvalues. In the same way, the power n of a matrix is obtained by taking its eigenvalues to the power n while keeping the eigenvectors.

The motivation for defining a pressure Hessian criterion comes from the small variation of pressure over the thickness of a boundary layer. It is considered that the number of layers in the boundary layer grid should be the same everywhere, to ensure the best grid quality [107]. These grid layers can be inserted on the original grid based on the approximate thickness of the boundary layer which is known. Therefore, it is unnecessary to employ a criterion which has very high values in the boundary layer region (e.g. velocity or vorticity-based). In addition, the pressure is important for the computation of forces on bodies and also can be considered a good indicator of waves and vortices.

Flux-component Hessian criterion (FCH). However, the difficulty of the pressure Hessian criterion to track wakes led to another criterion which adapts the mesh to pressure-based flows but also to boundary layers, wakes, and shear layers. This criterion is based on the Hessians of both the pressure and the velocity [108]. To give equal importance to the different Hessian matrices, a common weight ρU is assigned to all the velocity Hessians, where $U = \sqrt{u_i \cdot u_i}$, and the criterion is chosen as:

$$C_{FCH} = \left(max\left(\|H(p)\|, \rho U \|H(u_i)\|\right)\right)^n . \tag{3.23}$$

The maximum of two tensors is computed using the approximative procedure defined by Wackers et al. [109].

Hessian evaluation. To compute the Hessian matrix of a numerical solution, second-derivative operators must be discretized. In ISIS-CFD, the main technique to compute the Hessian is a smoothed Gauss (SG) method. For this method, first the gradient of a field q is computed using the Gauss approximation,

$$\nabla_G(q) = \frac{1}{V} \sum_f q_f S_f \mathbf{n}, \tag{3.24}$$

where the face values q_f are computed with the central discretization. Then, each component of the gradient is smoothed by applying several times a Laplacian smoothing $\mathcal{L}$,

$$\mathcal{L}(q) = \frac{\sum_f q_f S_f}{\sum_f S_f}. \tag{3.25}$$

In the next step, the Hessian matrix, the gradients of the smoothed gradient components, is computed using Eq. (3.24) and is symmetrized by:

$$(H)_{ij} = \frac{1}{2}((H)_{ij} + (H)_{ji}). \tag{3.26}$$

By applying several times the Laplacian smoothing (Eq. 3.25) to each component, the smoothed Hessian matrix is obtained.

A hybrid approach to compute the Hessian matrix is also possible. For the FCH criterion, the gradients are computed with the Gauss approximation (Eq. 3.24). Then, for numerical convenience, the derivatives of these quantities are computed with a least-squares approach.

3.3.3 Safety features

In ISIS-CFD, several measures are available to improve the regularity of the adaptively refined meshes.

Minimum cell size. A measure which is set by the user before starting a computation is to impose a minimum cell size: cells smaller than this size are no longer refined. This option prevents spurious refinement if locally large errors appear in the computation of the refinement criterion, which may appear for example in the high aspect-ratio cells of the near-wall boundary layer grid. Also, it prevents infinite refinement around flow singularities.

Boundary layer protection. The second measure is a protection of the boundary layer grid, which on the original grid consists of several layers of wall-aligned cells. For these layers, the refinement method contains an option to forbid any refinement in the wall normal direction (which would locally increase the number of layers). Furthermore, in all cases the refinement parallel to the wall is made the same in each column of cells from the wall to the outer layer (Figure 3.8): if one of the cells in a column needs to be refined, all the cells are refined. Thus, the column / layer structure of the boundary layer grid is preserved.

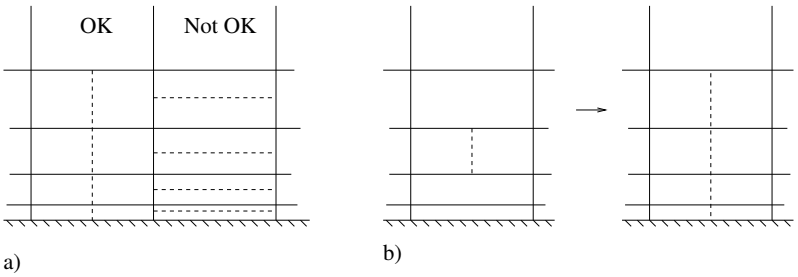
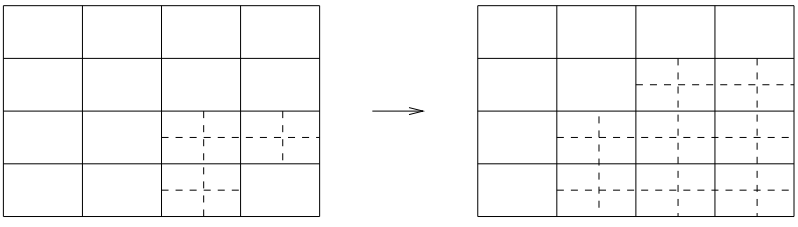
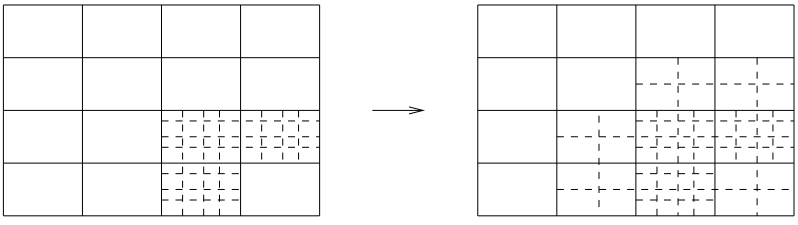


Figure 3.8 – Boundary layer protection: preventing normal refinement (a) and copying parallel refinement in a column (b)

Buffer layers. Buffer layers refer to cells that are refined around the zone where the criterion requires the mesh to be refined. Since the solution is evolving, either because a steady solution is converging or because the flow is unsteady, to obtain a margin of safety that allows the solution to change without having crucial flow features leave the zones of refined cells, the refinement process in ISIS-CFD is accompanied by introducing layers of safety cells. These safety cells normally have the same cell sizes as those specified by the criterion (Figure 3.9a). However, when the refinement criterion is nearly discontinuous, purely following the criterion would produce a grid that goes abruptly from very coarse to very fine cells. Therefore, to have cell sizes that gradually go from coarse to fine, the safety cells in the buffer layers are refined to a coarser size than those produced by the criterion, to get a gradual variation of cell size (Figure 3.9b).



(a) Standard buffer



(b) Fractional buffer

Figure 3.9 – Two types of buffer layer: refined mesh without and with application of the buffer

Refinement limiting boxes. This is an option which gives the possibility to limit the regions where the automatic refinement is active. Using limiting boxes, the grids are adapted only inside these boxes and the original mesh is kept intact for the rest of the domain. This method reduces the computational cost by concentrating the refinement in regions of interest and helps with imposing outflow conditions, by damping out flow features before the outflow face.

3.4 Use in this thesis

In this thesis, different features of the ISIS-CFD solver are used. In most cases, the solver is employed for computing single-fluid flow in stationary domains, so the multifluid capability and the mesh motion through U_d (Section 3.1) are not used. However, a ship in motion in a multi-fluid environment is studied in Chapter 7. Throughout the thesis, all the DES-based turbulence models (DES, DDES and IDDES) are used for various test cases. RANS computations are mostly based on k-ω SST, except for Chapter 7 where EARSM is selected as the statistical turbulence model. In addition, the AVLSMART scheme is used as the default discretization scheme, except for the DDES simulation of the ship in motion where a blended discretization scheme is applied.

The adaptive refinement is based on the flux-component Hessian (FCH) criterion, see Section 6.3. In Chapter 7, due to the presence of the free-surface, a combined FCH and free-surface criterion is used. Moreover, for each case an appropriate minimum cell size for the adaptation process is selected, particularly in DES-based computations whose highly unsteady solutions may stimulate the refinement criterion and lead to excessive refinements.

Chapter 4

Hybrid RANS/LES and mesh changes

hybrid RANS/LES turbulence models are sensitive to the grid size. For example, Detached Eddy Simulation (DES, see Section 2.2) as a hybrid RANS/LES model should act in RANS mode for boundary layers and act in LES mode for flows like separation where the momentum-carrying turbulent eddies are not much smaller than the scale of the geometry [90]. There is a special concern about excessively refined grids in regions where no destabilizing physical effect (like separation) exists. These fine grids, which are called the *ambiguous* grid type by Spalart [93], may activate the grid-dependent DES limiter (Eq. 2.18) and switch the model from RANS to LES. However, due to the lack of a physical destabilization, the production of velocity fluctuations may not be sufficient to compensate for the loss of modelled turbulent stresses due to the reduction of the eddy viscosity. This leads to the phenomenon which is referred to as modelled-stress depletion (MSD). MSD occurs in grid convergence studies, when the grid is gradually refined from a coarse mesh, and also when geometry features require a fine grid. Furthermore, adaptive grid refinement (AGR) may result in an extremely fine grid in which the turbulence model suffers from MSD.

An additional difficulty when combining AGR with the DES turbulence model is that grid adaptation provokes cell size variations in time and space. Thus, frequent local refinement creates a mesh which has different resolutions at different locations of the domain. In addition to the change of the turbulence model behavior due to its grid size sensitivity, the resolved turbulent content is affected by this type of grid. This behavior is also seen in full LES simulations in which there is a sharp interface between coarse and fine cells [14, 40, 74, 101]. When the flow goes from a fine to a coarse mesh, the fine LES structures are not resolved any longer by the coarse grid, so the resolved turbulence is reduced. In opposite, when the mesh becomes finer, the resolved content is not created spontaneously after the interface. Thus, the turbulence content cannot necessarily be transferred correctly through an interface.

This chapter studies these two issues. It starts by investigating the sensitivity of the DES turbulence model with respect to the mesh size for a turbulent flow over a flat plate (Section 4.1). For this test case, the performance of the turbulence model with respect to creating LES content and the possibility of MSD occurrence are considered. In Section 4.2, an artificial way of creating LES content is added: a synthetic method to generate unsteady inlet fluctuations is described and used in the flat plate test case to introduce artificial unsteadiness into to the boundary layer, with the aim of stimulating the DES model to produce more turbulence and therefore increase the resolved solution. The local cell-size dependency of the DES turbulence model is studied in Section 4.3, by considering a fixed

coarse-fine interface in a freely decaying turbulence test case. Finally, the chapter concludes with a discussion of the effect of mesh changes on a hybrid RANS/LES turbulence model like DES.

4.1 Grid sensitivity study for DES simulation

The effect of changes in the mesh on the performance of the DES turbulence model is investigated for an attached turbulent boundary layer over a flat plate, which is a severe case for the DES model when the fine grids extend into the boundary layer [93]. By refining the mesh in the streamwise direction, a convergence study is performed for an attached boundary layer where no physical mechanism (such as a separation) exists to stimulate the generation of unsteady LES content. In other words, the aim of the test is to intentionally create MSD.

In this test case, the turbulent boundary layer develops spatially over a flat plate which is located at $Y = 0$. The computational domain sizes in the streamwise, wall-normal and spanwise directions are $L_x = 0.5m$ 0, $L_y = 0.1m$, and $L_z = 0.005m$ respectively. A coarse mesh with $\Delta x^+ = 300$ and $\Delta z^+ = 30$ based on the friction velocity at $X = 0.4$, is generated for this test case. For the first layer Y^+ is set to 0.2; the thickness of the first cell in the wall-normal direction becomes $1.59167 \times 10^{-6}m$ and the cells are stretched with a ratio of 1.1 until to the top boundary. This coarse *Mesh 1* is refined in the streamwise direction to generate finer meshes ($Mesh$ 2-3). Morover, the stretching ratio in the wall-normal direction for the *Mesh 3* is increased to 1.2, so there is less number of grids in this direction. The major parameters of these meshes are gathered in Table 4.1. These meshes are chosen in a way to fall into Spalart's definition of an ambiguous mesh. They are not coarse enough in streamwise direction to be treated only by RANS, and also not fine enough to have a full LES computation.

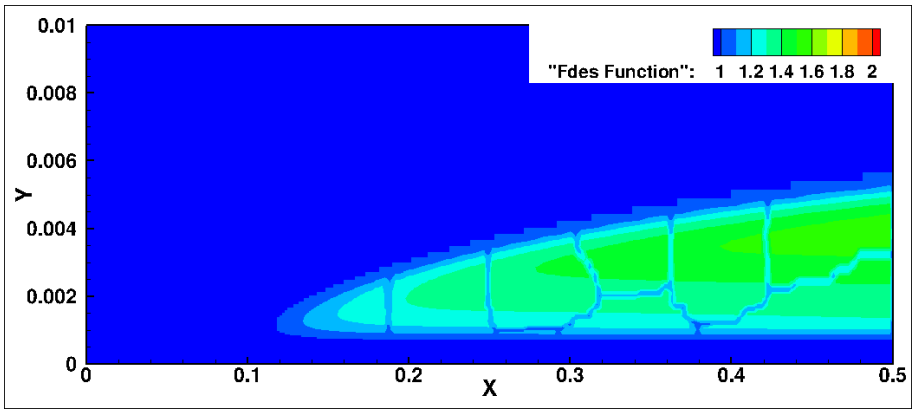
Table 4.1 – Parameters of the meshes employed in the flat-plate computation

Mesh name	$N_x \times N_y \times N_z$	Δx^+	Δz^+	N_{cells}
Mesh 1	$300 \times 137 \times 50$	300	30	2 055 000
Mesh 2	$800 \times 137 \times 50$	150	30	5 480 000
Mesh 3	$1500 \times 79 \times 50$	50	30	5 925 000
Mesh 4	$1000 \times 79 \times 100$	100	100	7 900 000

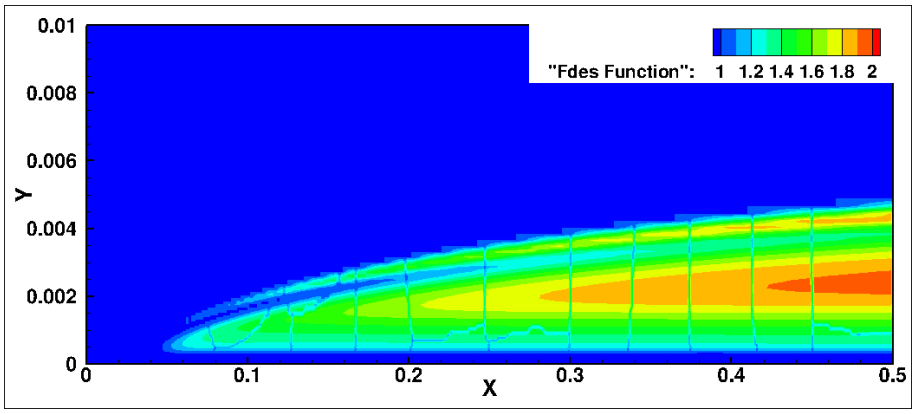
Periodic boundary conditions are used for the two spanwise boundaries and a no-slip condition is imposed on the wall. The velocity of the far field boundary at the inflow and top is equal to the free stream velocity ($U_e = 70\ ms^{-1}$). A uniform pressure is imposed for the outflow. The kinematic viscosity is $1.0 \times 10^{-6}\ m^2s^{-1}$ leading to a Reynolds number per meter $Re = 4.72 \times 10^6\ m^{-1}$. The computation is treated as unsteady and the time step for all the computations is set to $10^{-4}\ s$ to keep the accuracy at an acceptable level as the Courant number is increased by the refinement.

An initial streamwise grid-size sensitivity study is carried out with the three grids $Mesh$ 1-3. As a result of varying Δx, the regions in which $F_{DES} > 1$ increase (Figure 4.1). The turbulence model is switched from RANS to LES for a larger region of the boundary layer, so the RANS layer between the wall and the LES region becomes thinner. In addition, on finer meshes the value of the F_{DES} blending function increases, especially for $Mesh$ 3. Since this increasing F_{DES}, is a factor in the dissipation term of the k transport equation in the DES turbulence model formulation (Eq. 2.22), the dissipation of the turbulence kinetic energy (TKE) increases. However, no resolved turbulence kinetic energy is created in these

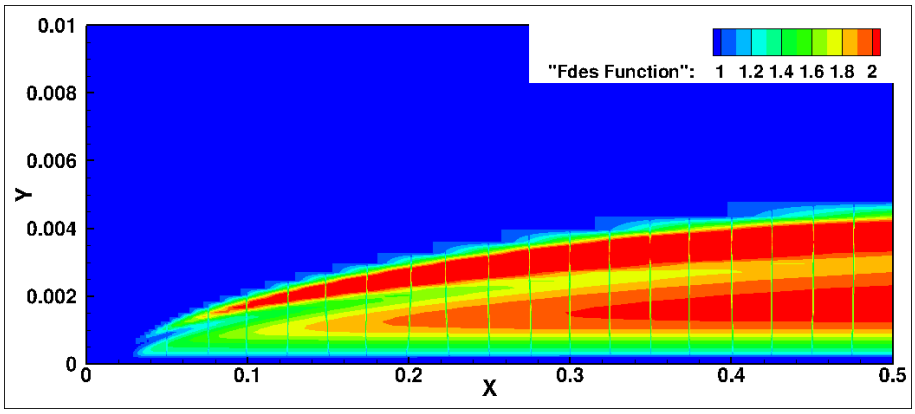
regions to counteract this reduction and therefore MSD occurs. Since the production of the resolved turbulence is zero, the solutions related to the resolved TKE are not given and only the reduction of the modelled TKE from *Mesh* 1 to *Mesh* 3 is presented in Figure 4.2.



(a) Mesh 1



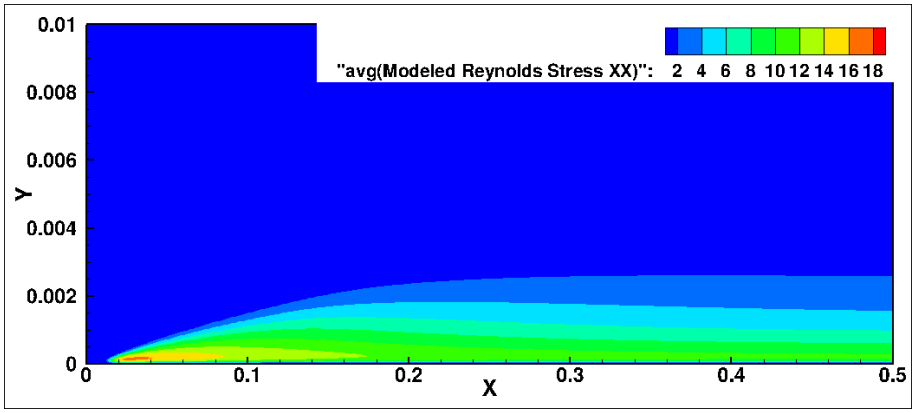
(b) Mesh 2



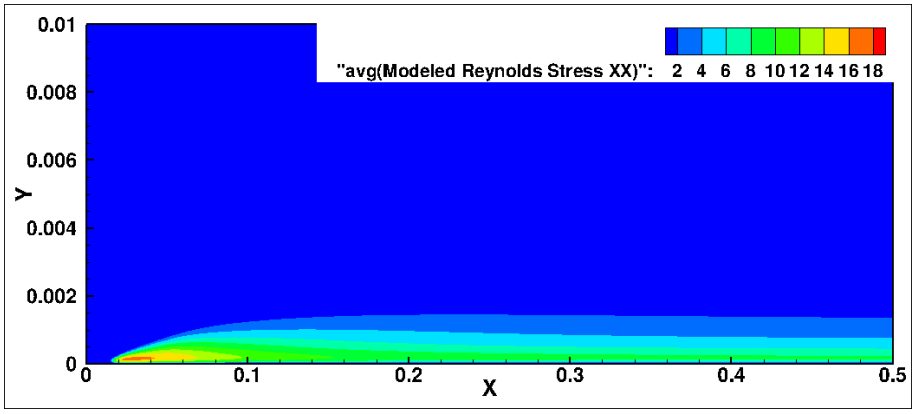
(c) Mesh 3

Figure 4.1 – F_{DES} function for the DES simulations over a flat plate with three different mesh resolutions in streamwise direction. The broken lines are due to a plotting issue

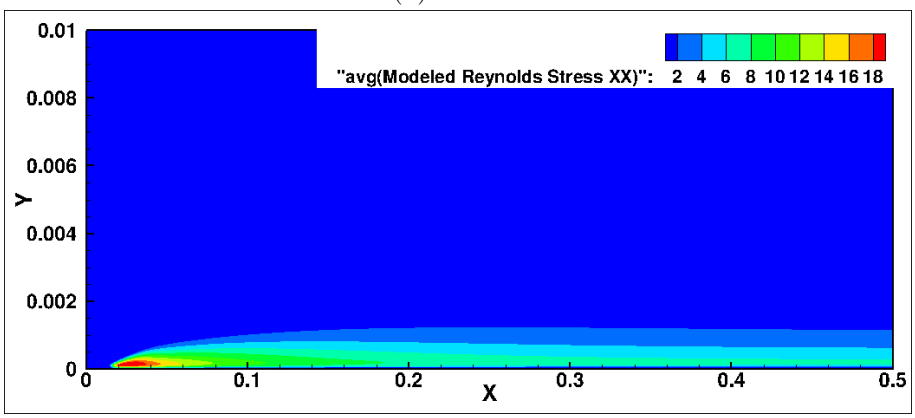
The averaged values of the total $u'u'$ turbulent stresses and the velocity profiles, at several cuts in the streamwise direction of the domain, are given in Figure 4.3. In the absence of the resolved solutions (they are all equal to zero), the total $u'u'$ turbulent stresses in Figure 4.3a are actually all modelled values and are not enough compared to the solution of the RANS computation. For attached turbulent flows like the flow over a



(a) Mesh 1



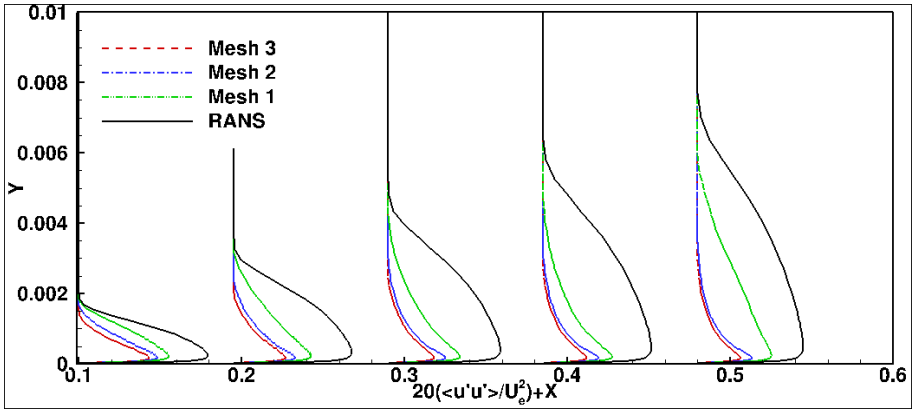
(b) Mesh 2



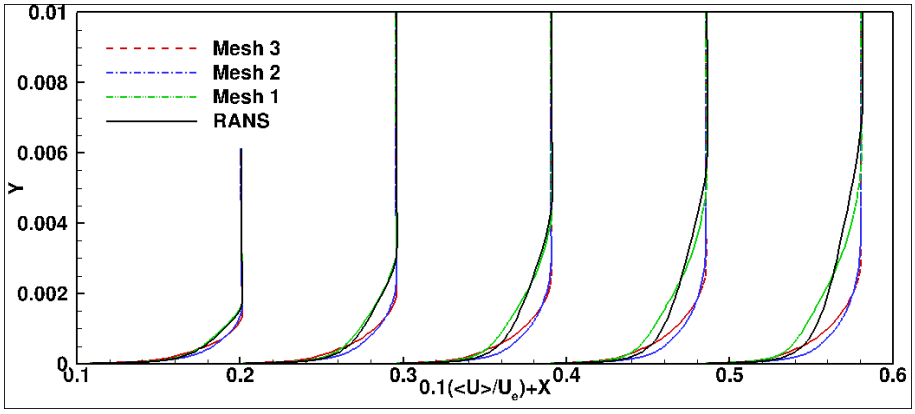
(c) Mesh 3

Figure 4.2 – Averaged modelled turbulence kinetic energy of the DES simulations over a flat plate with three different mesh resolutions in streamwise direction

flat plate, RANS is considered as a reliable model, so this is a suitable reference solution. The insufficient modelled solution is also reflected in the velocity profiles (Figure 4.3b) and the boundary layer thickness of these DES simulations. These results confirm that the solutions get far from the RANS solution as the resolution of the mesh in increased by refining in the streamwise direction.



(a) Averaged $u'u'$ turbulent stress



(b) Averaged streamwise velocity

Figure 4.3 – Averaged velocity profiles and turbulent stress at $X = 0.1, 0.2, 0.3, 0.4$ and $0.5m$ with three different mesh resolutions in streamwise direction

Since in the above simulations, the size of the domain in the spanwise direction is almost equal to the boundary layer thickness (δ) of the simulation $Mesh$ 1 at $X = 0.5$, the 3D effects of the turbulence, which may create resolved content, could be underestimated. Thus, another domain with a larger size in spanwise direction is tested. The computational domain size in the spanwise direction is set to $L_z = 0.05m$ which is around 10δ, while for the streamwise and the wall-normal directions the same size as the previous domain is chosen ($L_x = 0.5m$ and $L_y = 0.1m$). The properties of the new mesh ($Mesh$ 4) are presented in Table 4.1.

For the new computational domain, the simulation $Mesh$ 4 is performed with the DES model as before. Furthermore, a reference RANS simulation is done for the new computational domain, which is compared with the solution of the DES simulation $Mesh$ 4.

The blending function F_{DES} for the DES simulation with $Mesh$ 4 (Figure 4.4) shows that the LES behavior of the DES turbulence model concerns the same regions of the boundary layer as the simulations $Mesh$ 1-3 (Figure 4.1). The smaller values of the blending function for $Mesh$ 4, compared to $Mesh$ 3, are due to the coarser cells of this mesh in the streamwise direction. The rest of the solution behaves more or less the same as

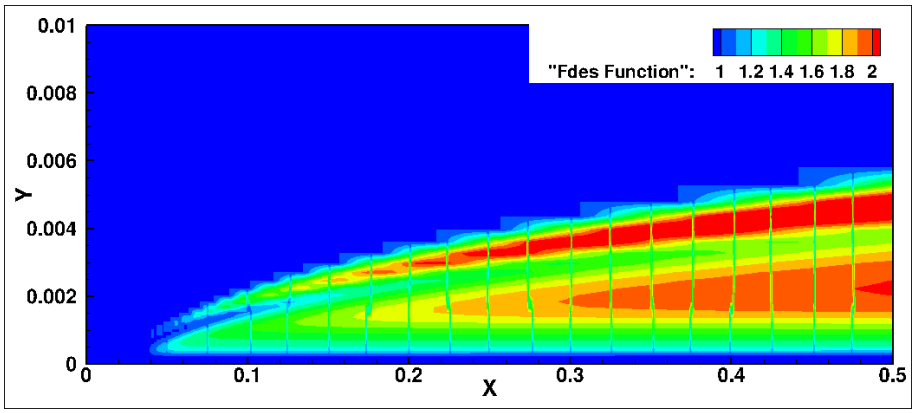


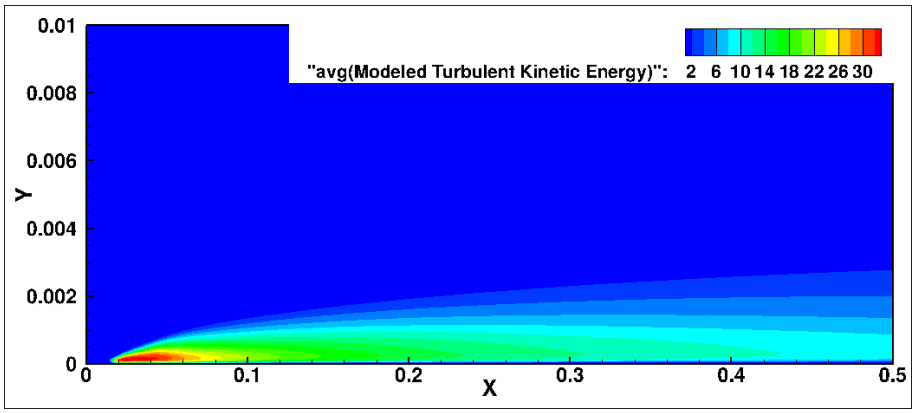
Figure 4.4 – F_{DES} function for the DES simulation $Mesh$ 4

the simulations with a thinner domain. The averaged turbulent stresses and velocity profiles for different locations (shown in Figure 4.10) are similar to the ones for the simulations $Mesh$ 2 and $Mesh$ 3.

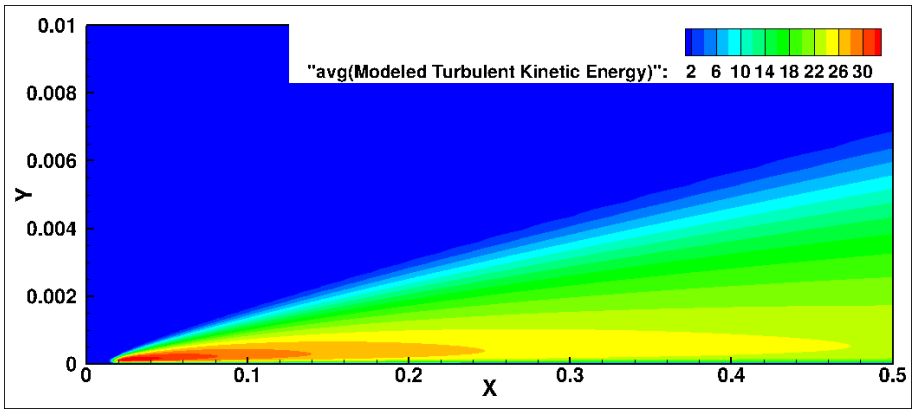
In Figure 4.5a, for the DES simulation, the averaged modelled turbulence kinetic energy in the region where the LES part of the turbulence model is activated, is low. However, the DES model cannot produce resolved turbulence to keep the turbulence balance. In contrary, the RANS simulation (Figure 4.5b) with a larger boundary layer thickness, has more modelled turbulence kinetic energy in almost the entire boundary layer, compared to the DES simulation. This confirms the existence of the MSD issue for the DES turbulence model. One solution for this issue is to avoid the activation of LES inside the attached boundary layer. DDES and IDDES (see Chapter 2) modify the DES limiter to have RANS for the entire boundary layer, so that all the turbulence in the attached boundary layers will be modelled, unless the mesh is fine enough to perform LES computations and a high-quality turbulence content is also provided at the boundaries. To test these modifications for the flat plate case, the IDDES turbulence model is used to perform the simulation for $Mesh$ 4.

In the simulation with the IDDES turbulence model, F_{DES} is zero everywhere (no figure is presented) and the entire domain is treated by RANS. Since $Mesh$ 4 is not fine enough for LES computation and no unsteady inlet boundary condition is used, the DDES branch of the model is activated and the model behaves like RANS. Comparing the averaged modelled turbulence kinetic energy (Figure 4.5 and Figure 4.10a) and the velocity profiles (Figure 4.10b) of this simulation with the results of the reference RANS simulation confirms that the simulation with the IDDES model is actually a RANS simulation, since the discussed prerequisite conditions for changing the turbulence model from RANS to LES inside the attached boundary layer are not met.

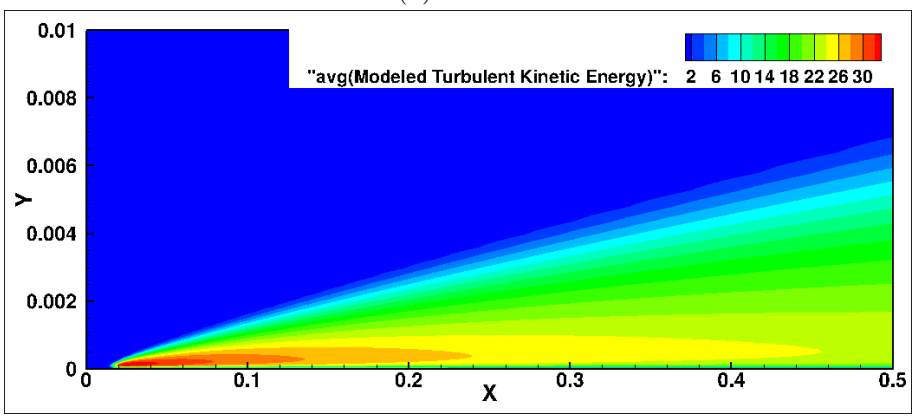
Thus, for the flat plate test case, the IDDES turbulence model works well because it behaves like RANS in the entire domain. In classical DES simulations however, the refinement causes a change in the behavior of the DES turbulence model inside the attached boundary layer. The model acts in LES mode in parts of the boundary layer where no physical mechanism exists to generate resolved content. Thus, the LES cannot produce resolved turbulence as a replacement for the reduced modelled turbulence and MSD occurs. A question is if, with artificial unsteadiness generated outside the boundary layer, the DES would be able to create resolved turbulence in the boundary layer, and the same result as the simulation with IDDES could be obtained. This question is studied in the next section.



(a) DES



(b) RANS



(c) IDDES

Figure 4.5 – Averaged turbulence kinetic energy for the simulations with $Mesh$ 4

4.2 DES simulation with unsteady inlet boundary condition

A real flat-plate boundary layer is unsteady, however in all the DES simulations discussed so far, no unsteadiness is created. One possibility which could be tested is imposing unsteadiness into the system using unsteady inflow fluctuations. Unsteady fluctuations at an inlet are usually used to impose LES-like turbulent flow at the inlet boundary of full LES simulations. The same technique is used here in order to initialize fluctuations in the region where the DES turbulence model acts in LES mode.

In this section, several approaches to generate inflow turbulence are described. Among these available approaches, the one which is reliable and most compatible with the ISIS-CFD solver is implemented and used for the DES simulation of the flat plate test case, to impose a synthetic unsteadiness inside the boundary layer.

4.2.1 Unsteady inlet fluctuations

In the LES mode of hybrid methods, the unsteady three-dimensional energy-carrying eddies are resolved. Hence, the velocity specified at the inflow of the computational domain should ideally represent the contribution of these eddies. The choice of the method to specify this inflow condition is problem-dependent and various techniques can be used:

- Inflow turbulence generation using a precursor calculation,
- Inflow turbulence generation by recycling,
- Synthetic inflow turbulence generation.

Precursor DNS or LES. For this approach, the velocity fluctuations at the inlet are set equal to normalized/rescaled fluctuations from a precursor DNS or well-resolved LES of some 'canonic' flow (e.g. a developed channel flow or a flat plate boundary layer) or from corresponding databases [87]. In practice, the velocity field in a plane normal to the streamwise direction is stored for each time step. The sequence of planes is then used as inflow data (e.g. unsteady Dirichlet conditions) for a separate calculation of the flow of interest [83]. Despite its accuracy, the method is rather computationally expensive in terms of both CPU and memory load. Other than that, the method is not self-sufficient (since it relies upon external databases) and the inflow data are restricted in terms of the Reynolds number, which makes its applicability to complex (far from the canonical) high Reynolds number flows questionable [87].

A variant of this method for hybrid RANS/LES simulations has been developed by Schluter, Pitsch and Moin [84] in the case where an LES domain is located downstream of a RANS domain. This method uses the mean velocity field from the RANS solution and adds turbulence extracted from a pre-generated database created by an auxiliary LES computation as follows:

$$u_{(i),LES}(x,t) = \bar{u}_{(i),RANS}(x) + u'_{(i),Database}(x,t), \qquad i = 1,2,3. \tag{4.1}$$

Recycling of turbulence. This is an approach where the simulation generates its own inflow conditions. The method consists in taking a plane at a location several boundary layer thicknesses δ downstream of the inflow. The data in this plane are then rescaled and reintroduced at the inflow [83]. This method was first proposed by Lund et al. [58] who apply these inlet conditions in pure LES and DNS.

The recycling-rescaling method was originally developed for flat plate boundary layers. The adaptation of the recycling process to more complex geometries is considered as one of the main difficulties of this approach. Another difficulty of the recycling method is related to the initialization of the recycling process. In general, a secondary turbulent boundary layer simulation (i.e. a database approach) has to be used to provide initial perturbations, before the recycling process can be started.

Synthetic inflow turbulence generation. Compared to recycling and precursor simulations methods, synthetic techniques, which imply superimposing some externally-generated artificial (synthetic) turbulent velocity fluctuations onto the (U)RANS velocity field at the upstream LES inlet, are currently considered as most suitable for the simulation of complex industrial flows [87].

Their attractive features include self-sufficiency, relative ease of implementation, computational efficiency, and last but not least, tolerance to grid type and topology. However, the artificial nature of the created turbulence often results in a rather long adaptation region. The accuracy of synthetic turbulence generators depends on the quality of the turbulence (its closeness to real turbulence), which in turn depends upon the specific properties of the synthetic turbulence generator used [87]. A classical approach for the generation of synthetic inflow data, which is described in detail in Section 4.2.2, is the synthetic random Fourier method (SRFM). In this method, a finite sum of discrete Fourier modes is used to generate a random velocity field. This turbulent three-dimensional field is isotropic and homogeneous [5].

4.2.2 Implementation of SRFM

Among all the available methods to generate inflow turbulence, the synthetic random Fourier method is widely used, reliable and can be easily implemented without any need for additional modification to the solver. In this method, the velocity field for the synthesized turbulence at one time step is given by N random Fourier modes as

$$\mathbf{v}'(\mathbf{x}) = 2\sum_{n=1}^{N} \hat{u}^n \cos\left(\boldsymbol{\kappa}^n \cdot \mathbf{x} + \psi^n\right) \boldsymbol{\sigma}^n, \tag{4.2}$$

$$\boldsymbol{e}_{\boldsymbol{\kappa}^n} = \begin{bmatrix} \sin(\theta)\cos(\phi) \\ \sin(\theta)\sin(\phi) \\ \cos(\theta) \end{bmatrix}, \tag{4.3}$$

where $\hat{u}^n$, ψ^n and σ_i^n are the amplitude, phase and direction of Fourier mode n. The direction of the wavenumber vector $\boldsymbol{\kappa}^n$ is determined by the unit vector $\boldsymbol{e}_{\boldsymbol{\kappa}^n}$ (Eq. 4.3) with the two random angles ϕ^n and θ^n (Figure 4.6). The magnitude κ^n of the wavenumber vector $\boldsymbol{\kappa}^n$ comes from a non-random wavenumber space between the smallest wave number κ_1 and the highest wave number κ_{max} which is divided into N modes, equally large, of size $\Delta\kappa = (\kappa_{max} - \kappa_1)/N$. In addition, the unit vector $\boldsymbol{\sigma}^n$, the direction of Fourier mode n, is denoted by the random angle α^n. Continuity requires that the unit vector, $\boldsymbol{\sigma}^n$ and $\boldsymbol{\kappa}^n$ are orthogonal (Eq. 4.4). The variables ϕ^n, θ^n, ψ^n, and α^n are random; ϕ^n, ψ^n, and α^n are selected in the range of $[0, 2\pi]$, and θ^n is in the range $[0, \pi]$.

$$\boldsymbol{\sigma}^n = \boldsymbol{R}^n \boldsymbol{e}'_{\boldsymbol{\sigma}^n} = \boldsymbol{R}_z(\phi^n)\boldsymbol{R}_y(\theta^n)\boldsymbol{e}'_{\boldsymbol{\sigma}^n} \quad with \quad \boldsymbol{e}'_{\boldsymbol{\sigma}^n} = \begin{bmatrix} \cos(\alpha) \\ \sin(\alpha) \\ 0 \end{bmatrix}, \tag{4.4}$$

where $\boldsymbol{e'}_{\boldsymbol{\sigma}^n}$ is the unit vector corresponding to the direction of the vector $\boldsymbol{\sigma}^n$ in the $X'Y'Z'$ coordinate (Figure 4.6). In addition, the rotation matrices $\boldsymbol{R}_y(\theta^n)$, $\boldsymbol{R}_z(\phi^n)$ are defined as

$$\boldsymbol{R}_y(\theta^n) = \begin{bmatrix} \cos(\theta^n) & 0 & \sin(\theta^n) \\ 0 & 1 & 0 \\ -\sin(\theta^n) & 0 & \cos(\theta^n) \end{bmatrix}, \quad \boldsymbol{R}_z(\phi^n) = \begin{bmatrix} \cos(\phi^n) & -\sin(\phi^n) & 0 \\ \sin(\phi^n) & \cos(\phi^n) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{4.5}$$

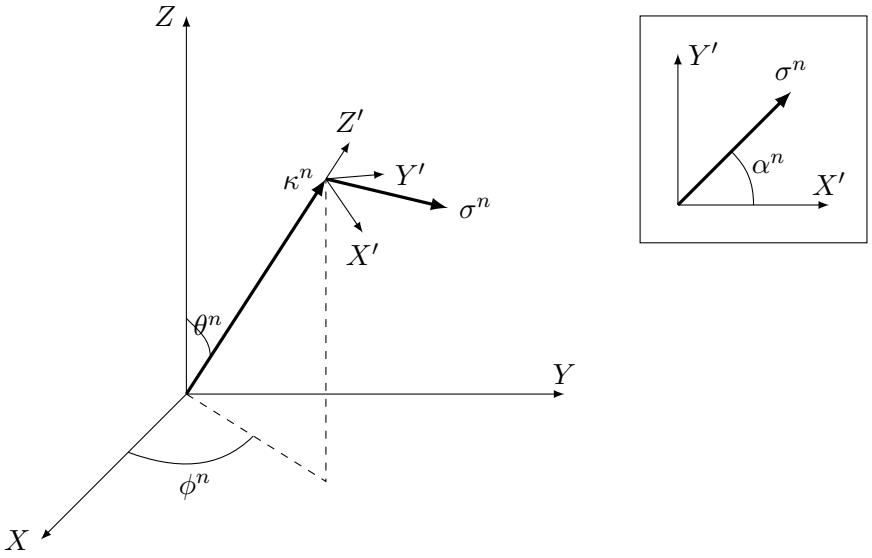


Figure 4.6 – Wave vector geometry for the nth Fourier velocity mode

The amplitude $\hat{u}^n$ of each mode in Eq. (4.2) is equal to $\sqrt{E(\kappa^n)\Delta\kappa}$, and one uses the modified von Karman spectrum $E(\kappa)$ to simulate the complete spectral range:

$$E(\kappa^n) = c_E \frac{u_{rms}^2}{\kappa_e} \frac{(\kappa^n/\kappa_e)^4}{\left[1 + (\kappa^n/\kappa_e)^2\right]^{17/6}} e^{[-2(\kappa^n/\kappa_\eta)]^2}. \tag{4.6}$$

In the above equation, κ_η corresponds to the Kolmogorov wave number which is defined by $\epsilon^{1/4}\nu^{-3/4}$. In addition, κ_e is the wave number where the spectrum has its maximum (Figure 4.7), defined by $\kappa_e = c_E 9\pi/(55L_t)$. The turbulent length scale L_t may be estimated in the same way as in RANS simulations: $L_t = C_\mu k^{1/2}/\omega$, where C_μ is a model constant which in the standard version of the k-ω model has a value of 0.09 [114]. The coefficient c_E is obtained by integrating the energy spectrum over all wave numbers to get the total turbulence kinetic energy and is equal to 1.453 [20].

The smallest wave number is defined by $\kappa_1 = \kappa_e/p$. The factor p should be larger than one to make the largest scales larger than those corresponding to κ_e. A value $p = 2$ is suitable [20]. The highest wave number is defined based on the mesh resolution: $\kappa_{max} = 2\pi/(2\Delta)$, where Δ is the grid spacing. Often the smallest grid spacing near the wall is too small, so a slightly larger value must be chosen.

The fluctuations are generated on a grid with equidistant spacing (or on a weakly stretched mesh), $\Delta Y = Y_{max}/N_2$, $\Delta Z = Z_{max}/N_3$, where N_2 and N_3 denote the number of cells in the Y and Z direction, respectively. The fluctuations are set to zero at the wall and are then interpolated to the inlet plane of the CFD grid (the Y-Z plane).

The fluctuating velocity fields which are generated in each time step are independent of each other and their time correlation will thus be zero. This is non-physical. To create a

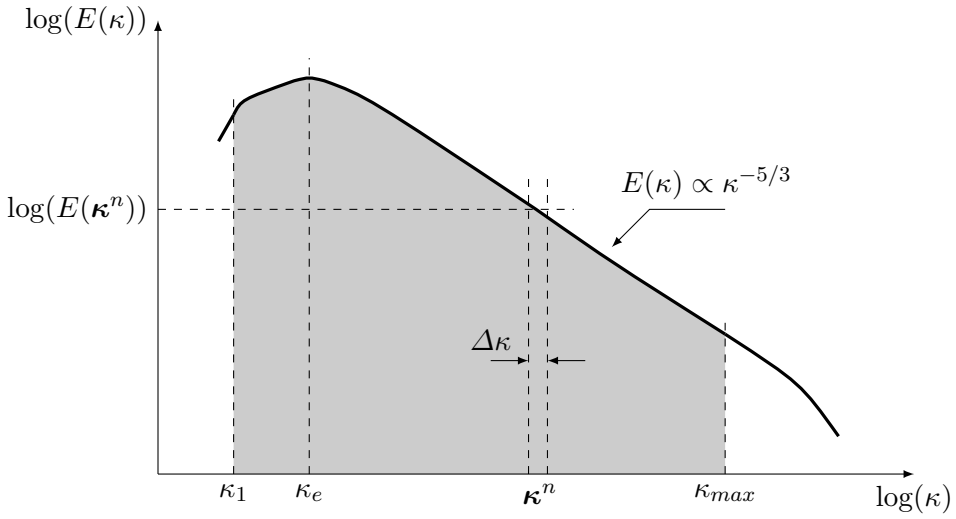


Figure 4.7 – Modified von Karman spectrum

correlation in time, a new fluctuating velocity field $\boldsymbol{v'}$ is computed based on an asymmetric time filter [20]

$$(\boldsymbol{v'})_m = a(\boldsymbol{v'})_{m-1} + b(\mathbf{v'})_m, \tag{4.7}$$

where m denotes the time step number and

$$a = \exp(-\Delta t/T_{int}), \tag{4.8}$$

where Δt and T_{int} denote the computational time step and the integral time scale, respectively. The second coefficient is taken as

$$b = (1 - a^2)^{0.5}. \tag{4.9}$$

Having $\hat{u}^n$, κ_j^n, σ_i^n and ψ^n, allows to compute the fluctuating velocity fields at the inlet. Finally, the computed velocity fluctuation fields are added to a constant inlet velocity field or, like in Eq. (4.1), can be added to a velocity field from a precursor RANS computation.

4.2.3 Flat plate with unsteady inlet boundary condition

The SRFM method described above is used in a DES simulation for the flat plate test case. The velocity fluctuations are generated before each time step and added to the constant inlet velocity. The unsteady inlet velocity in the streamwise direction for several instants of the computation is presented in Figure 4.8. The fluctuation is in the range of $\pm 1\%$ of the inlet stream velocity (70 m/s). The stretching of the fluctuation contours in the Y-axis is due to the stretch of the grids from the flat plate to the top boundary in the wall-normal direction.

The DES simulation with $Mesh$ 4 is repeated with the unsteady inlet boundary condition (named DES(+UBC)). The first effect of adding unsteady fluctuations to the inlet velocity is on the blending function. Figure 4.9a shows that the blending function is larger than one for a larger zone and therefore the area which is treated by LES is wider. Inside the boundary layer, the value of F_{DES} becomes larger compared to the DES simulation without the unsteady velocity at the inlet boundary (Figure 4.4). In addition, Figure 4.9c shows that turbulence is resolved in part of the LES region due to the unsteadiness which is introduced in the boundary layer by the small inflow fluctuations. However, comparing with

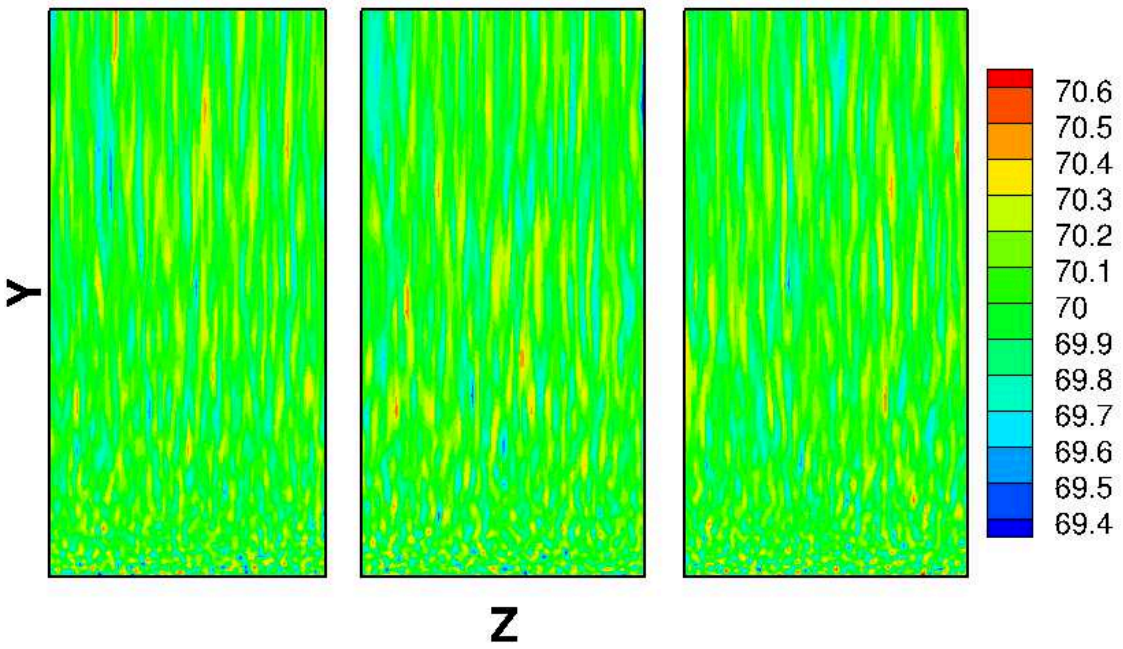
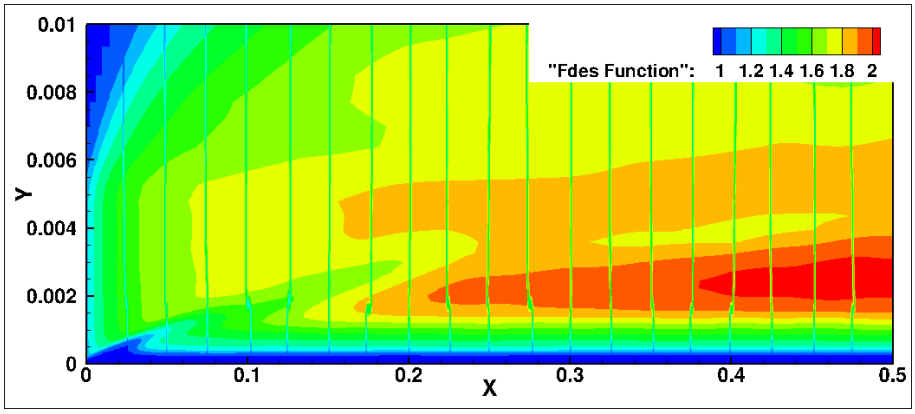


Figure 4.8 – Inlet generated velocity (U_x) at 3 instants with intervals of $20\Delta t$

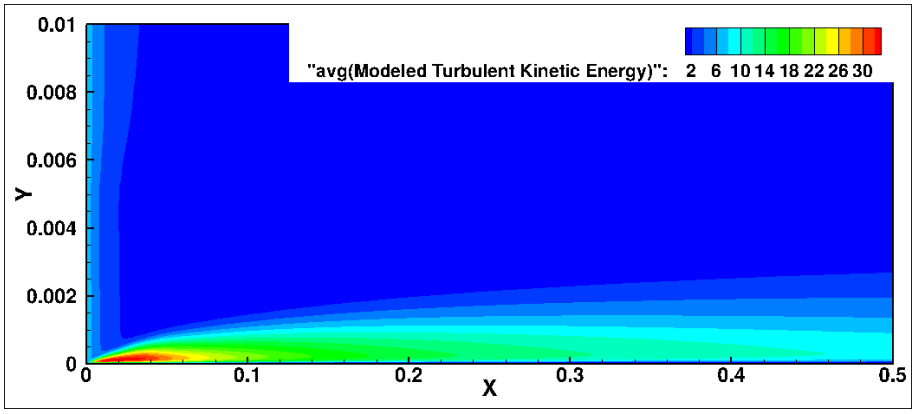
the modelled turbulence kinetic energy of the RANS or IDDES simulation in Figure 4.5, it is seen that these resolved solutions are not enough to compensate for the decrease in the modelled turbulent solutions due to the turbulence model change from RANS to LES. Moreover, the boundary layer thickness at the end of the domain remains the same as the simulation without the unsteady inlet fluctuations ($\delta = 3.3 \times 10^{-3}\ m$) and the averaged velocity profiles (Figure 4.10b) at different locations are almost similar. These results indicate that while the artificial fluctuations successfully impose unsteadiness within the boundary layer, this is not enough to balance the lack of the modelled turbulence and to get accurate results.

For the DES simulation with unsteady boundary condition, the effect of changing the time step is also studied, since the unsteady eddies may be diminished by a too large time step. In addition to $\Delta t = 10^{-4}$, two more time steps are also tested ($\Delta t = 2{\times}10^{-4},\ 5{\times}10^{-5}$). For these two time steps, Figure 4.11 shows slight differences in the averaged resolved TKE compared to the simulation with time step $\Delta t = 10^{-4}$ (Figure 4.9c). The profiles of the averaged total turbulent stress also confirm the changes in the solutions (Figure 4.12a). However, the convergence of the solutions when changing the time step is not monotone. This randomness in the solution may be caused by the unsteady inlet fluctuations since, when the time step is changed, the speed of updating the fluctuations changes as well. This can have an effect on the resolved turbulence which is created by the DES model. Despite the slight detected changes in the turbulence kinetic energy, the averaged velocity profile remains intact for all the three simulations with different time steps (Figure 4.12b).

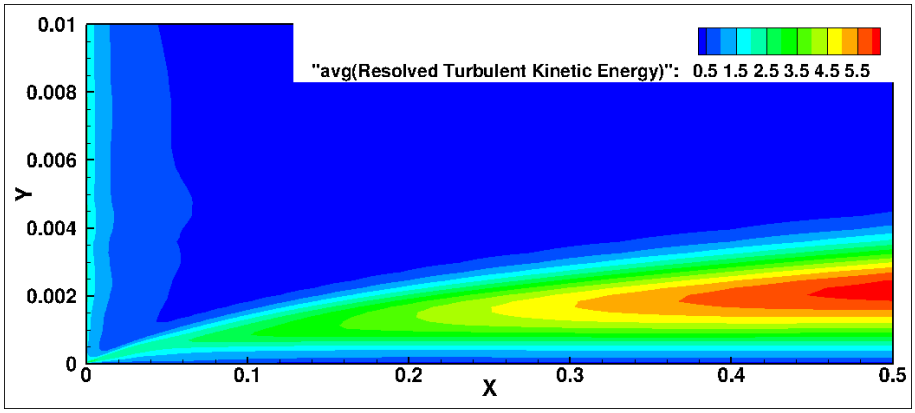
Thus, the unsteady fluctuations at the inlet boundary do not lead to an effective increase in the resolved turbulence. However, they are amplified inside the boundary layer, they increase the unsteadiness and activate the LES mode of the turbulence model for a larger region. This implies that if the boundary layer is physically unsteady, the DES model should be able to capture the unsteady eddies. The turbulence for the flat plate case comes from the friction at the vicinity of the wall and spreads inside the boundary layer. But the RANS region close to the wall models this turbulence instead of simulating it as unsteadiness and does not transmit it to the resolved eddies in the LES part. The results



(a) Blending function F_{DES}

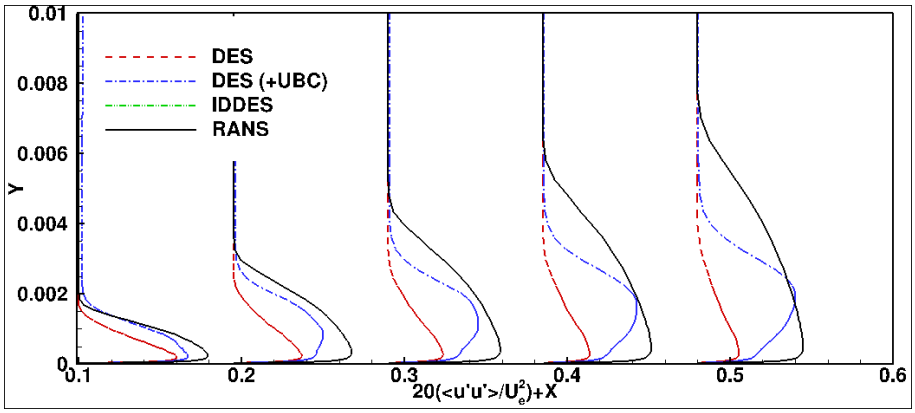


(b) Averaged modelled TKE

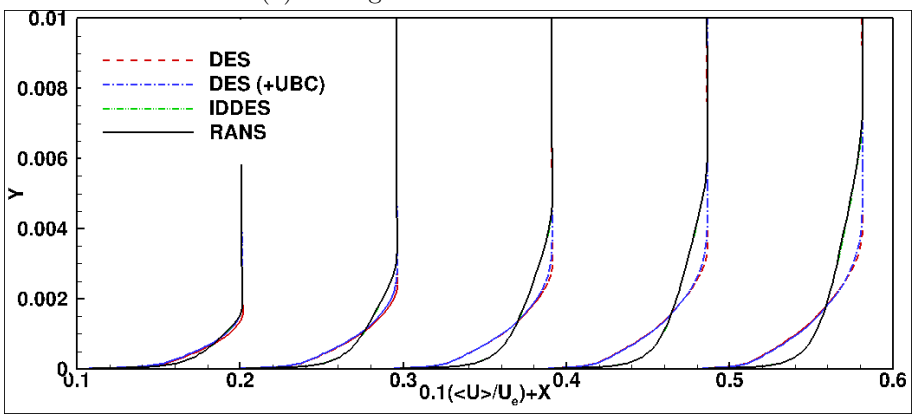


(c) Averaged resolved TKE

Figure 4.9 – Blending function F_{DES} and the averaged TKE for the DES simulation with unsteady inlet boundary condition - different scales are used for the modelled and resolved TKE to have a better representation



(a) Averaged $u'u'$ turbulent stress



(b) Averaged streamwise velocity

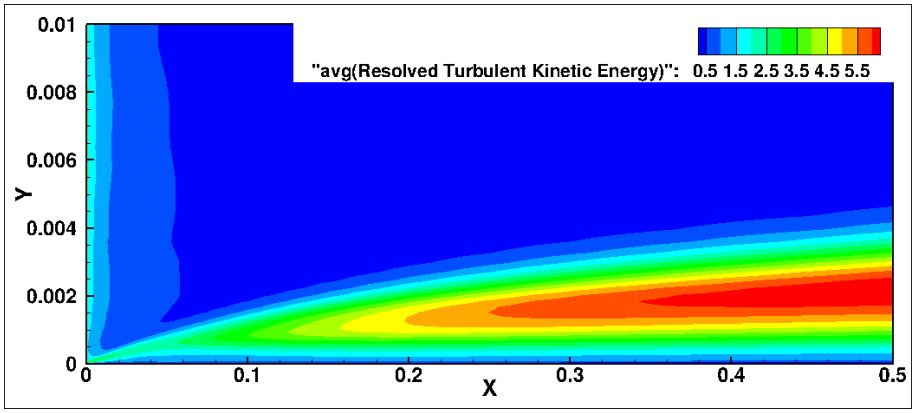
Figure 4.10 – Averaged velocity profile and turbulent stress for the simulations with $Mesh$ 4 in the streamwise direction at $X = 0.1, 0.2, 0.3, 0.4$ and $0.5m$

show that with this effect missing, a small outside perturbation cannot generate enough turbulent structures on its own. So unless the boundary layer is resolved with full LES, refining the mesh will just increase the MSD, deteriorating the quality of the solution.

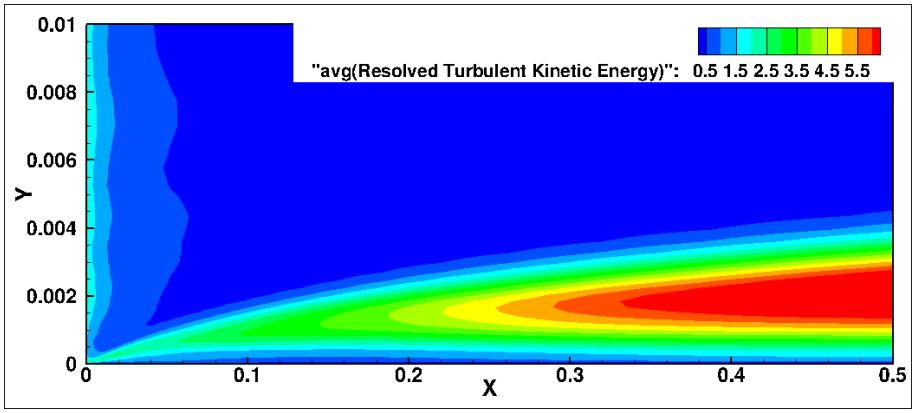
4.3 Coarse/Fine interface

Another problem of the combination of a hybrid RANS/LES turbulence model with AGR, is the effect of an interface between coarse and fine cells on the transmission of the turbulence through that interface. Grid adaptation varies locally the cell size in time and space. As a result, a mesh with different local resolutions is created which affect the resolved turbulent content - with possibly a switch of the turbulence model. These effects depend on the flow direction. For instance, at a coarse to fine interface, the blending function F_{DES} increases. Thus, the modelled turbulence is reduced due to a decrease in the turbulent viscosity. However, resolved content is not spontaneously created to compensate for the loss of the modelled solution. On the contrary, for a fine to coarse interface, fine upstream resolved turbulence is dissipated at the interface since it cannot be represented on the coarser mesh, and it is not directly replaced by modelled turbulence. This may violate the turbulence balance between the upstream and downstream side of the interface and perturb the solutions.

In this section, to study the effect of a coarse/fine interface on the performance of a hybrid RANS/LES turbulence model, a freely decaying turbulence case in a steady uniform



(a) Δt=0.00005

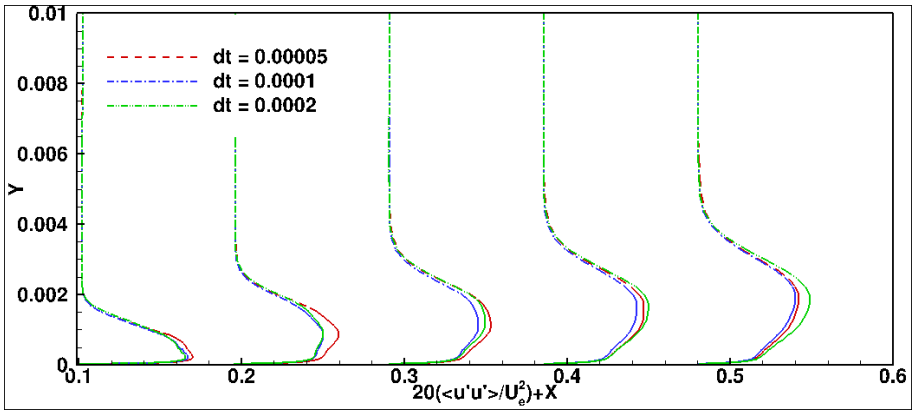


(b) Δt=0.0002

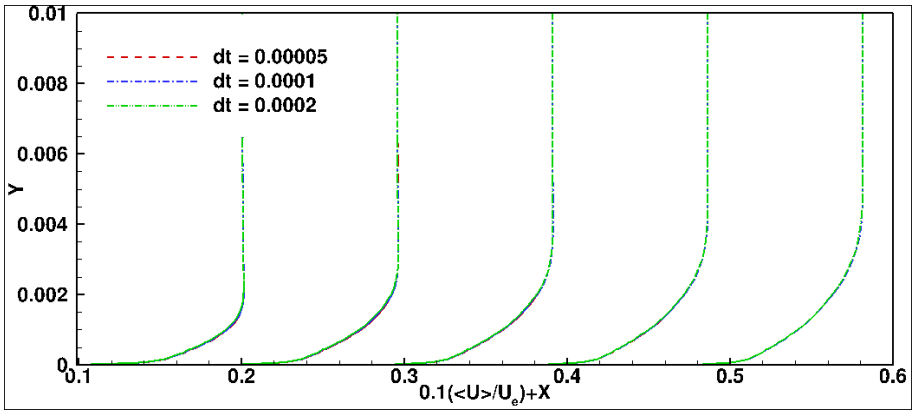
Figure 4.11 – Time-step effect on average resolved turbulence kinetic energy

Table 4.2 – Mesh properties for the free decaying turbulence simulations

Simulation name	Upstream Resolution	Downstream Resolution	Grid
Uniform (Fine)	192^3	192^3	Uniform
Uniform (Coarse)	48^3	48^3	Uniform
Fine $\Rightarrow$ *Coarse*			
FC-Coarse	48^3	24^3	Non-Uniform
FC-Medium	96^3	48^3	Non-Uniform
FC-Fine	192^3	96^3	Non-Uniform
Coarse $\Rightarrow$ *Fine*			
CF-Coarse	24^3	48^3	Non-Uniform
CF-Medium	48^3	96^3	Non-Uniform
CF-Fine	96^3	192^3	Non-Uniform

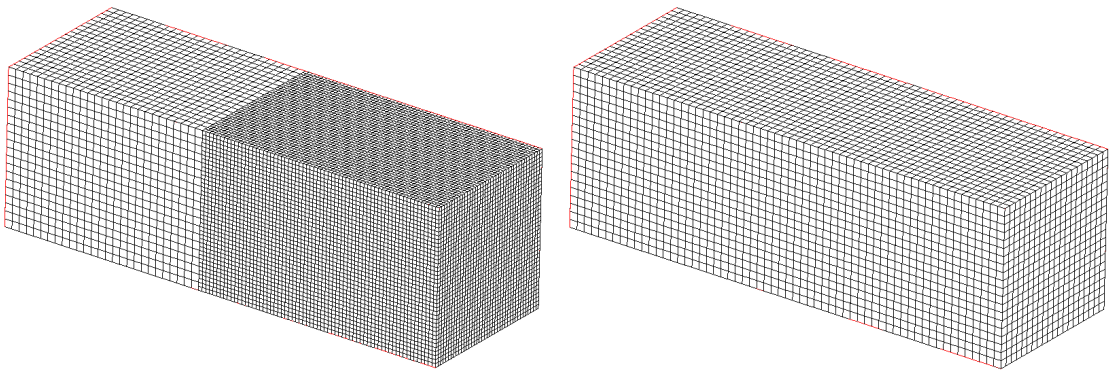


(a) Averaged $u'u'$ turbulent stress



(b) Averaged streamwise velocity

Figure 4.12 – Time-step effect on velocity profile and turbulent stress for the simulations with $Mesh$ 4 at $X = 0.1, 0.2, 0.3, 0.4$ and $0.5m$



(a) Non-uniform (b) Uniform

Figure 4.13 – Free wall domain and meshes for the free decaying turbulence

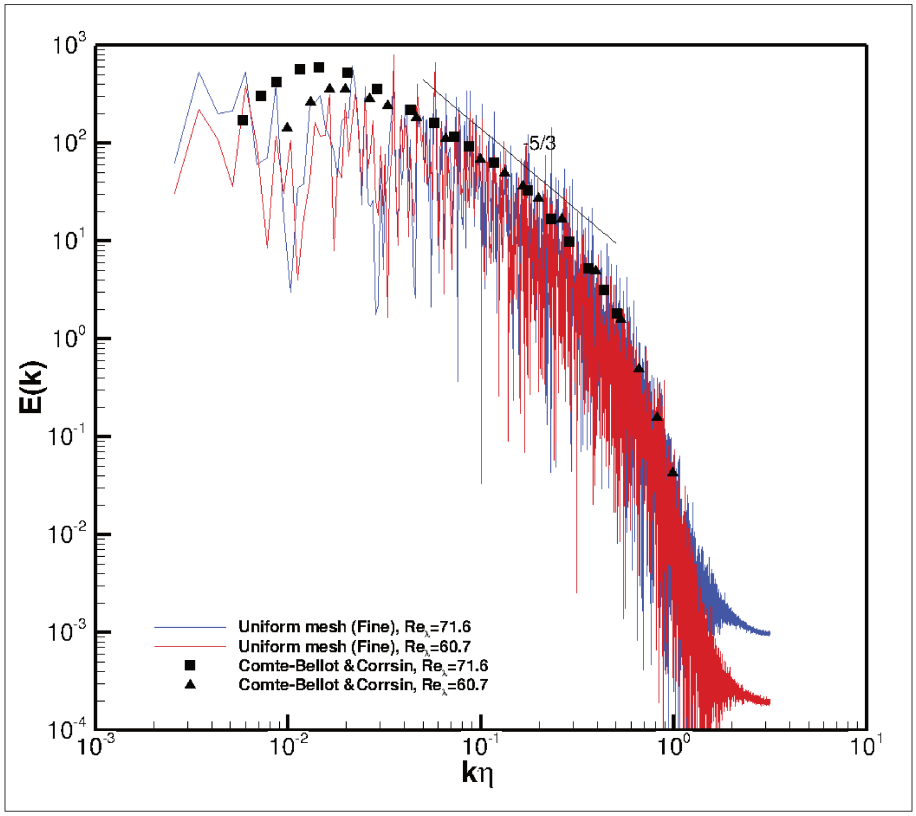
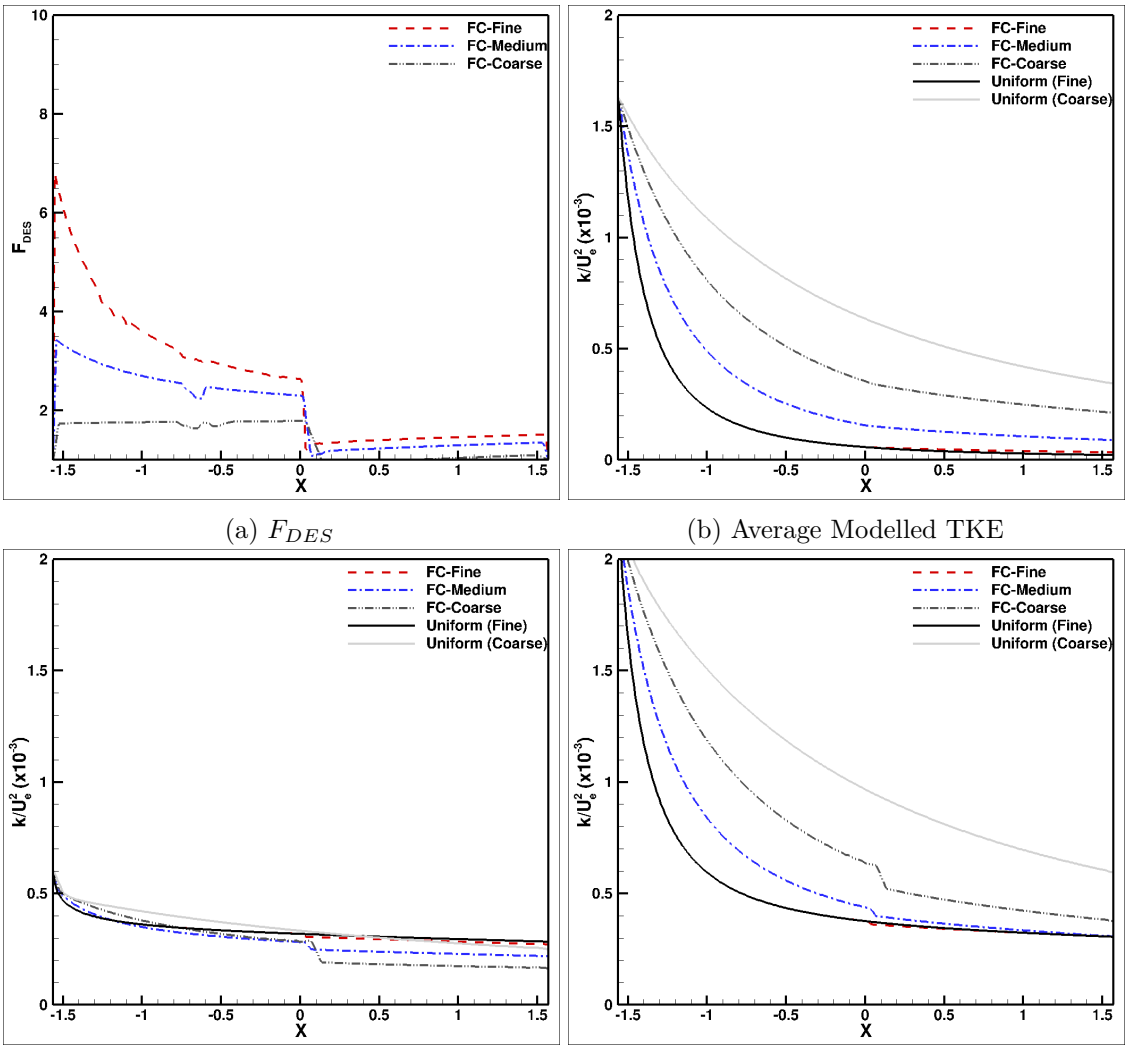


Figure 4.14 – Energy spectrum of the uniform meshes at $R_\lambda = 60.7$ and 71.6, compared with time-decaying experiments

flow is considered. It is inspired by the work of Goodfriend et al. [39] on the assessment of the turbulence decaying across a grid refinement interface in LES simulations. The whole domain consists of two equal parts with a dimension of $0.05\pi \times 0.05\pi \times 0.05\pi$ for each part. Each part has a uniform mesh resolution which can be either finer or coarser than the other part (Figure 4.13a) or similar to the resolution of the other part (Figure 4.13b). Table 4.2 represents the properties of the uniform and non-uniform meshes which are used for simulating the free decaying turbulence. The effect of the interface between the two parts of the domain is studied by two scenarios: the first one is to have the finer mesh in the upstream part and study how the interface affects the solutions as the flow passes through a fine to coarse interface (FC). In the second scenario, the coarse part is upstream and a coarse to fine interface (CF) is considered.

For all the simulations, the freestream velocity is $U_e = 70ms^{-1}$. Unsteady velocity fluctuations are generated and added to the inlet velocity according to the synthetic method discussed in Section 4.2.2. The initial Taylor microscale Reynolds number (Re_λ) based on the root mean square of the velocity fluctuations ($\langle V' \rangle_{rms}$) and the Taylor microscale ($\lambda = \sqrt{15\nu/\epsilon}\langle V' \rangle_{rms}$) is about 285. Mirror boundary conditions are used for the four boundaries in the Y and Z directions and a uniform pressure is imposed on the outflow. The computation is unsteady and the time step is set to $10^{-4}s$.

While the main purpose of these simulations is to assess the production and the dissipation of turbulence across the grid refinement interface, the uniform-grid results are compared with experiments first. However, any experimental data or DNS solution concerning the spatial decaying rate for steadily convected turbulence at this relatively high initial microscale Reynolds number, to the best of the author's knowledge, is not available. Therefore, the energy spectra of two uniform-grid simulations at $Re_\lambda = 71.6$ and



(a) F_{DES}

(b) Average Modelled TKE

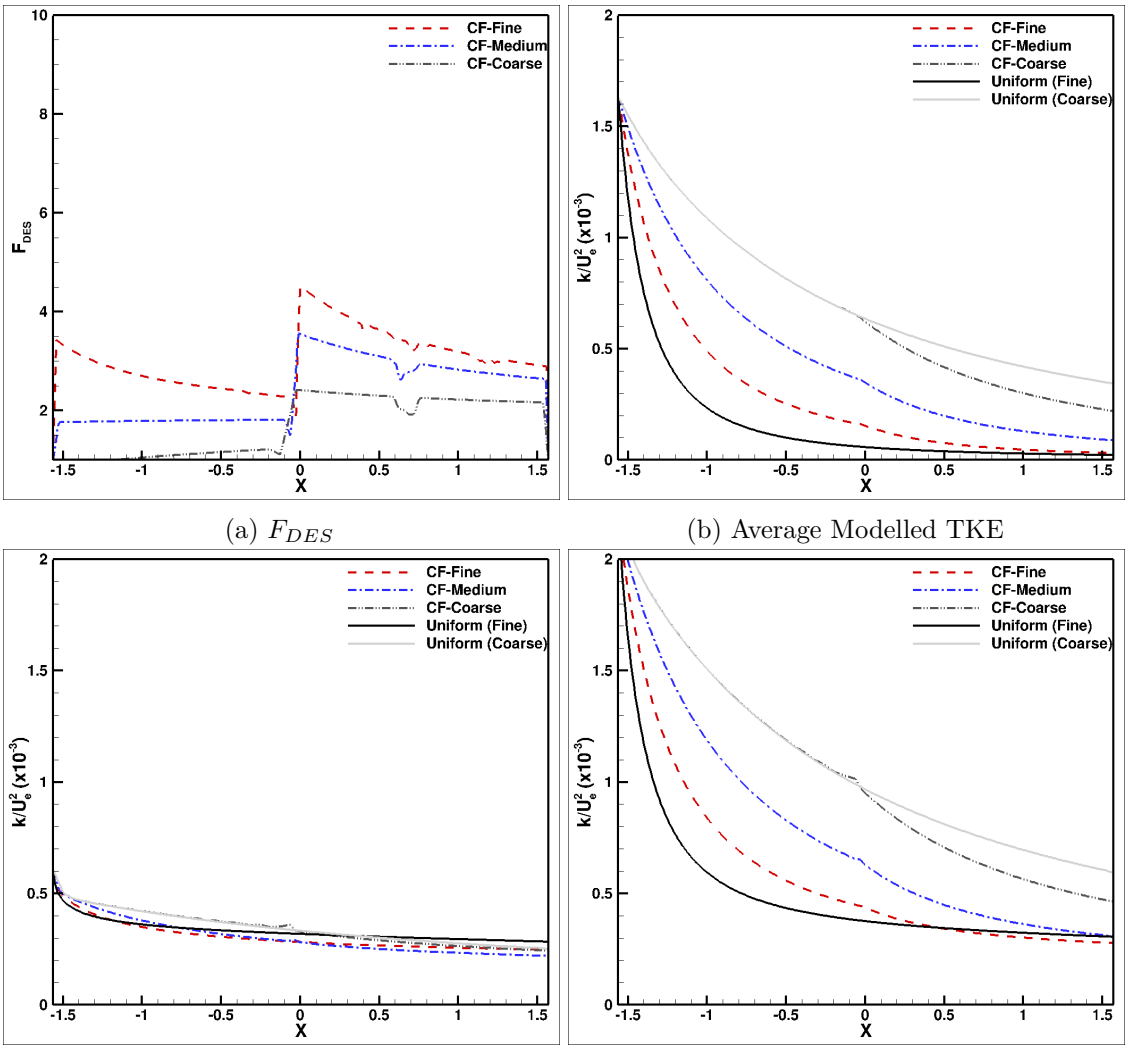
(c) Average Resolved TKE

(d) Average Total TKE

Figure 4.15 – F_{DES} and non-dimensionalized averaged turbulence kinetic energy for the simulations with fine-to-coarse transition (FC)

60.7 (Figure 4.14) are compared with the time decaying rate of uniform free turbulence in an experiment done by Comte-Bellot and Corrsin [19]. The results are reassuring since the solutions of DES simulations match the Kolmogorov energy spectrum relation. The difference between the computational results and the experiment is due to the lower intensity of the turbulent fluctuations at the inflow of the domain.

For the main computations, the decay of the turbulence kinetic energy along the streamwise direction is presented in Figures 4.15 and 4.16. For each coordinate in streamwise direction, the turbulence kinetic energy shown is averaged over the cut plane normal to the streamwise direction. The changes in the mesh size have an effect on the resolved turbulence at the interface. For the scenario in which the flow passes from the fine to the coarse grid, the resolved turbulence is reduced (Figure 4.15c) since the finest turbulent structures on the fine mesh cannot be resolved on the coarse mesh. This difference in the resolved TKE before and after the interface is larger for the simulations with coarser meshes, especially for the simulation *FC-Coarse* which has a sharp drop. This effect is large enough to affect the total averaged turbulence kinetic energy in Figure 4.15d. In contrast, when going from coarse to fine (Figure 4.16c), all the resolved turbulence on the coarse grid can be



(a) F_{DES}

(b) Average Modelled TKE

(c) Average Resolved TKE

(d) Average Total TKE

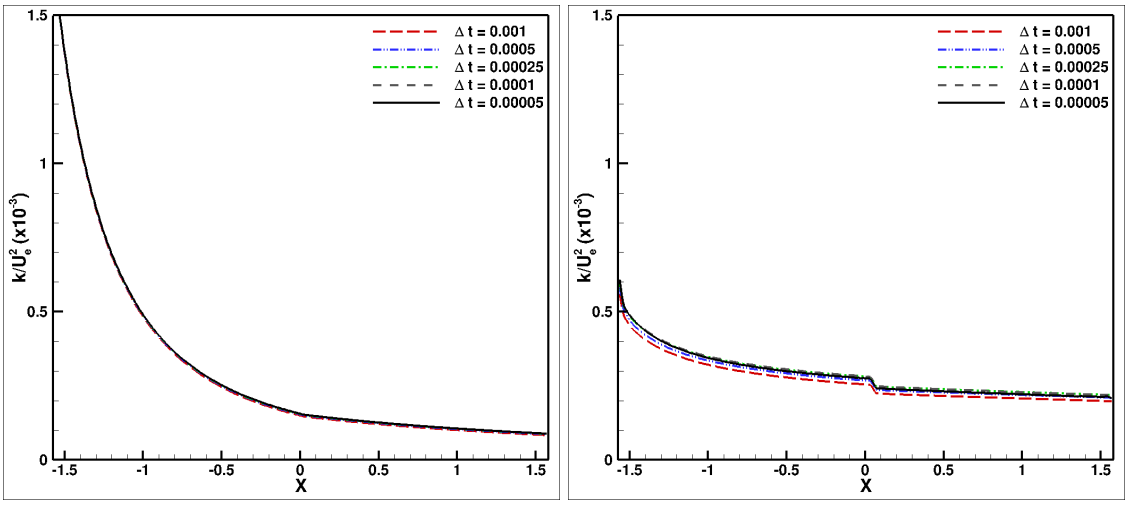
Figure 4.16 – F_{DES} and non-dimensionalized averaged turbulence kinetic energy for the simulations with coarse-to-fine transition (CF)

preserved by the downstream fine grid, so there is no jump at the interface, except for the CF-$Coarse$. For the simulations with fine and coarse uniform meshes, regardless of the dissipation rate difference between the two simulations, no discontinuity in the solutions is seen. These perturbations in the solution as a result of the fine and coarse interface were also detected by Piomelli et al. [74] and Goodfriend et al. [39] while performing large eddy simulations in which interfaces between fine and coarse grids were artificially introduced.

In addition to the sudden perturbation at the coarse/fine interface, a change in the grid size varies the rate of the dissipation of turbulence. In Figures 4.15a and 4.16a, regardless of which scenario is considered, the value of the blending function F_{DES} in the finer part of the domain is larger than in the part with low-resolution grids. A sudden decrease (increase) happens at the interface from fine to coarse (coarse to fine) cells. Figure 4.15b shows that for the finer parts, the modelled turbulence decays faster. This is due to the high value of the F_{DES} which results in a higher dissipation rate, so after the interface depending on the change of the local grid size, the turbulence decay increases (finer grids, higher F_{DES}) or decreases (coarser grids, lower F_{DES}). In contrast, the resolved TKE is better preserved on finer grids due to lower numerical dissipation (Figure 4.15c). However, since there is no

source of instability, the model is not triggered to create resolved turbulence. In general, the dependency of the modelled turbulence on the mesh is huge and the change in resolved turbulence is not enough to counterbalance this dependency. These effects are the same for the two scenarios, it does not matter if the coarse grid comes first, or the fine grid.

The influence of time step The time step effect on the rate of the turbulence decay is investigated as well. According to Figure 4.17 which corresponds to *FC-Medium*, the change in the dissipation of the modelled TKE as a result of the change in the time step is less than for the resolved solution. For other simulations, the behavior is more or less the same and the the modelled TKE can be considered as time-step independent. Even the changes in the resolved TKE are less than the differences between meshes, which means that the spatial resolution is the dominant factor in the behavior of the turbulence.



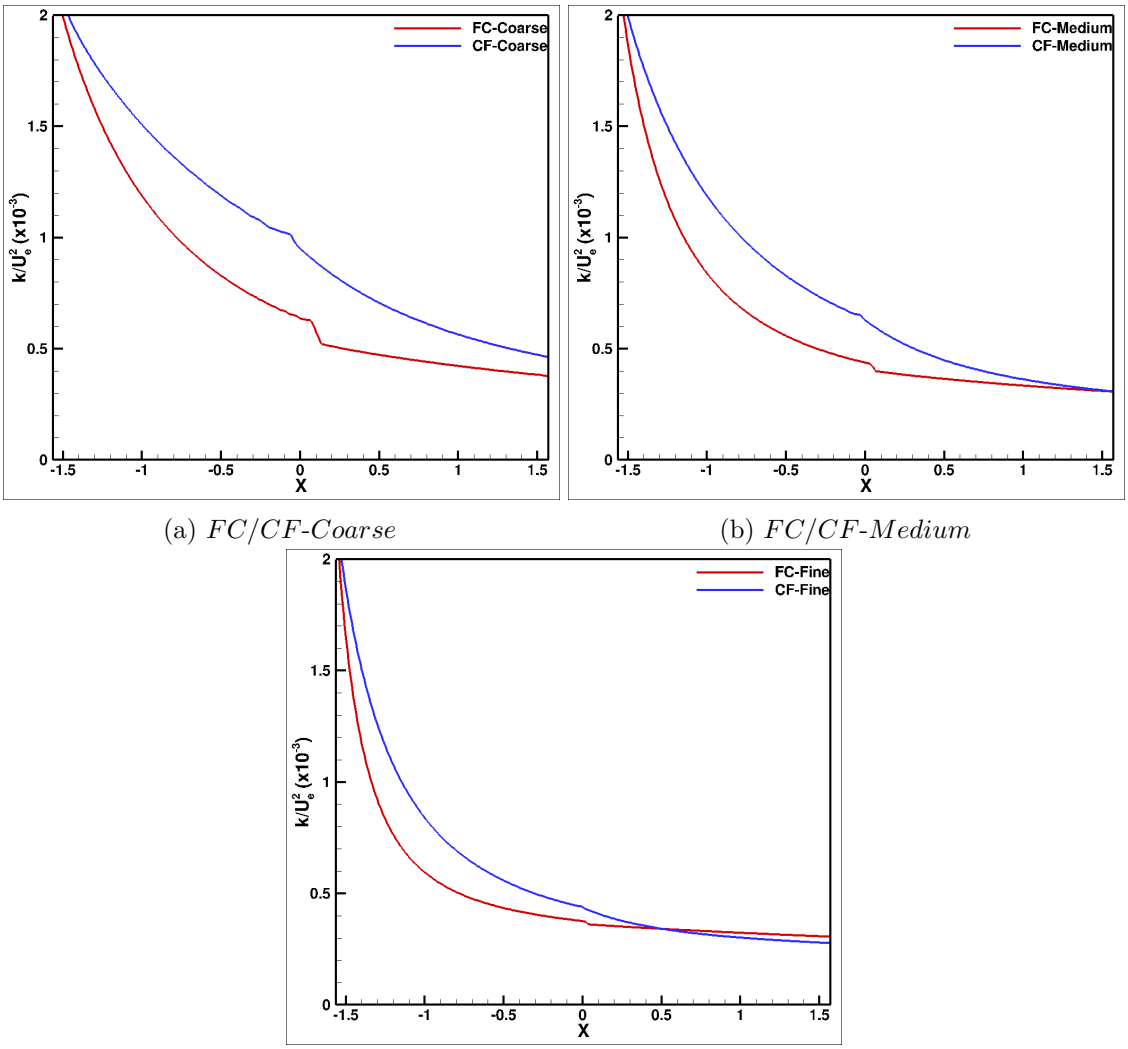
(a) Average Modelled TKE (b) Average Resolved TKE

Figure 4.17 – Time-step effect on the turbulence kinetic energy for the simulation *FC-Medium*

In addition, in Figure 4.18 for each mesh, the averaged total turbulence kinetic energy for the simulations FC ($Fine \Rightarrow Coarse$) and CF ($Coarse \Rightarrow Fine$) are compared. The figure shows that, while the difference between the results decreases on finer meshes, the turbulence decay in DES simulations is highly mesh-dependent. It is possible that if the modelled and resolved TKE at the inflow were adjusted to the grid resolution at the inlet, the results on coarse and fine meshes would be closer. According to the principle of DES, on finer grids the amount of inflow modelled TKE should be reduced and the amount of resolved TKE should be increased and the way around for the coarser meshes. However, in realistic conditions, this argument does not hold since in an unstructured mesh, all kinds of cell sizes may exist in the domain and the mesh is not necessarily perfect for the flow at all given positions. In addition to the current result that when a grid size interface occurs, the changes in resolved turbulence do not compensate for those in the modelled turbulence, this leads to the conclusion that any grid size change must be seen as a perturbation for DES simulations when modelled and resolved turbulence coexist.

4.4 Conclusion

In this chapter, the performance dependency of DES as a hybrid RANS/LES turbulence model on the mesh resolution and also the effect of the interface between fine and coarse



(a) *FC/CF-Coarse* (b) *FC/CF-Medium*

(c) *FC/CF-Fine*

Figure 4.18 – Non-dimensionalized average total turbulence kinetic energy

grids are investigated. The study of a turbulent flow over a flat plate shows that the DES turbulence model is sensitive to mesh changes. By refining the mesh in streamwise direction, the LES behavior of the DES turbulence model is activated for a larger region inside the boundary layer. For these regions the turbulent viscosity is reduced and therefore the modelled turbulence decreases. However, no resolved solution is produced to replace the reduction of the modelled solutions, therefore modelled stress depletion occurs. The distance between the DES solutions and the reference RANS computation increases as the mesh is refined. This is due to the lack of unsteadiness inside the boundary layer which should be generated by a physical instability in the flow. The conditions for instability in the flat plate boundary layer are normally created by the friction of the flow with the wall, which produces a shear flow. However, the effect is suppressed by the RANS layer close to the wall. Although using synthetic velocity fluctuations at the inlet boundary is successful in imposing unsteadiness inside the LES regions and in stimulating the DES model to produce resolved turbulence, the generated turbulence is not enough to avoid MSD. Thus, one should be cautious when using fine meshes inside the boundary layer if this can cause a coexistence of RANS and LES zones. Either the whole boundary layer should be treated by RANS, like the IDDES simulation in which the DDES branch of the

model is activated inside the boundary layer and the model behaves as RANS, or all the conditions of a wall modelled LES (WMLES) simulation (see Section 2.2), like a fine mesh and realistic inlet fluctuations, must be present.

In addition, the effect of an interface between coarse and fine cells in freely decaying turbulence is studied. Two separate effects of the grid size change are detected: (1) the loss of the resolved TKE at the interface when going from fine to coarse grids, since the coarse mesh cannot resolve the fine turbulence structures coming from the fine mesh. (2) the change in the dissipation rate of the modelled turbulence behind the interface as a result of a difference between the values of the blending function F_{DES} on the two side of the interface. For instance, for the higher-resolution meshes, the blending function is larger and the turbulence decays faster. Thus, mesh changes in space (and by extension in time) affect the behavior of the DES turbulence model and perturb the solution by changing the production and dissipation of the turbulence, which results in a violation of the turbulence balance. Therefore, it can be implied that less mesh changes and locally uniform meshes are ideal for the DES turbulence model.

Due to the effects discussed above of the grid size change on the behavior of the DES model, problematic behavior can occur in a simulation using adaptive grid refinement, whenever a coarse mesh is refined or a high-resolution mesh is locally coarsened. Although the improved versions of this turbulence model, such as DDES and IDDES, reduce the dependency of the model on the local grid resolution, the coupling of AGR with DES-type turbulence models must be done with great care, since regardless of excessive refinement or coarsening which can push the turbulence model to its limits, any grid refinement interface imposes perturbations to the system which deteriorate the solution.

Chapter 5

Analysis of solution averaging

The complexity of AGR and hybrid RANS/LES coupling. Like other hybrid RANS/LES turbulence models, detached eddy simulation (DES) is sensitive to cell size changes [68] and its performance is affected as the mesh changes. In Chapter 4, it was shown that the transition from a coarse mesh to a fine one imposes perturbations like the depletion of modeled turbulence without the generation of sufficient LES content, so-called *modeled stress depletion* (MSD), or the opposite, the loss of the resolved solution during the transition from a fine mesh to a coarse one. Over the years, models like DDES or IDDES have been proposed to reduce the effects of these issues (see Chapter 2).

However, when adaptive grid refinement (AGR) is used, the refinement or derefinement of cells may also provoke these problems. For instance, at the beginning of a simulation, the local cell sizes are large, so that the length scale of the hybrid RANS/LES model corresponds to the RANS model. AGR then refines the mesh gradually in the locations where flow features demand a fine grid. The locally refined mesh which activates the DES limiter ($\tilde{d} = C_{DES}\Delta$, Eq. 2.18) may not be fine enough to support the resolved LES content. Moreover, even if the mesh becomes fine enough, there is not necessarily a physical mechanism to create the resolved turbulence immediately. As a result, the eddy viscosity and therefore the modeled Reynolds stress is reduced, without any sizable resolved stress to compensate for this loss. In the opposite situation, when the local grids are derefined and the mesh becomes too coarse to resolve small eddies, parts of the resolved turbulence in the solutions is removed without replacement. And when the turbulence model is switched from LES to RANS, it is not sure that all resolved turbulence is transformed to modeled turbulence. Thus, the changes in mesh size which are inherent in AGR may deteriorate the performance of the turbulence model.

In addition to the turbulence modeling issue, the AGR has another perturbative effect: the truncation error of the interpolation between coarse and fine grids. AGR tries to follow all the turbulent structures of the flow by refining the mesh in places where the second derivatives of the flow components are non-zero. Therefore, in unsteady flows where the solution is always changing, using AGR means intensive mesh changes every few time steps. These mesh changes that could be repeated several times for each location of the computational domain, are accompanied by the interpolation of the solutions from the old to the new grid. This large number of interpolations may eventually result in a considerable error and therefore perturb the solution.

Averaging window. All these discussed issues make the use of adaptive grid refinement together with a hybrid RANS/LES turbulence model complicated. A possible solution to this problem is to use the adaptive refinement in such a way that the changes of the mesh are limited, in order to reduce its effect on the performance of the turbulence model and

to remove the interpolation errors as much as possible. To change the adaptive refinement, three parameters are available: the threshold, the minimum cell size and the criterion (see Chapter 2). The threshold parameter is set for the whole mesh and does not control the refinement locally. The minimum cell size limits the refinement but only for the smallest cells. Therefore, the right choice is to work with the refinement criterion which determines which part of the mesh should be refined. In ISIS-CFD, the refinement criterion tensor in the AGR process is usually defined from the instantaneous velocity and pressure fields at the refinement time step (Chapter 3). The refinement is then performed based on this computed criterion. Therefore, limiting the mesh changes is possible by limiting the changes in the refinement criterion. Since the instantaneous solution evolves fast, particularly for highly unsteady flows, the criterion which is computed from this solution, is influenced by its evolution and changes. The proposed option here is to compute the criterion based on an averaged solution instead of the instantaneous solution. This restricts the refinement to capturing only the main unsteady flow features, without following all the turbulent structures.

In AGR based on an averaged flow field, the averaging interval used to compute the refinement criterion is of great importance. The averaging interval can be chosen from a short sliding averaging window to the entire computational time, based on the flow behavior and the desired grid refinement approach. For adaptive refinement, a long interval means a slow-changing mesh because fluctuations in the flow need a lot of time to change the averaged solution. This results in a time-lag of the refinement response to the flow evolution. In the particular case of (quasi-)periodic flows, long intervals lead to a slow convergence of the mesh. However, after a reasonable amount of computational time, the mesh changes reduce to the extent that the mesh can be assumed constant. In contrast, shorter intervals make the refinement more reactive. Since the averaged solution evolves faster and therefore the mesh changes faster, more dynamic eddies are followed by the refinement. With short intervals, it is not possible to obtain a converged mesh for (quasi-)periodic flows but selecting an appropriate interval can limit the fluctuations in the averaged solution and therefore reduce the mesh change, with the difference that the mesh refinement reacts faster to any significant change of the flow.

The necessity of flow physics study. In order to have an idea about the right size of the averaging interval, the physics of the flow should be considered. It is important to know what unsteady features exist in a flow and which one is dominating. Then based on their frequencies, it can be decided which ones will be averaged out and which ones will be followed by the adaptive meshing.

As an example of such a study, in this chapter, a turbulent separated flow behind a backward-facing Step (BFS) is considered. It is widely used in most of the research on the evaluation of hybrid RANS/LES models [41, 42, 86]. Before using this test case, the behavior of this class of fluid motion especially from the standpoint of time dependency is described in Section 5.1. Then in Section 5.2, the backward-facing step case is simulated with a reference fine mesh with no adaptive refinement, and the results are compared with the available experimental data. In Section 5.3 an analysis is carried out over the signals of several point probes to study the frequencies of the different features which exist in the flow. Then the effect of different averaging intervals on the averaged solution is studied (Section 5.4). The chapter ends with a conclusion which is presented in Section 5.5.

5.1 Backward-facing step

The backward-facing Step (BFS) flows, also called "sudden expansion flows", are flows between parallel planes that face an expansion of the bottom plane. The expansion is known as the step. The BFS flow has attached boundary layers upstream of the step and on the upper wall of the channel, a separation which starts from the fixed separation point (the tip of the step) and a reattached boundary layer on the step wall, which inherits turbulent content from the upstream separation region. The BFS is a basic model which involves the most important features of separated flow that can be seen in daily life, such as airfoils at large angle of attack, separation flows behind a vehicle, and also the flow around a ship in a static drifting motion.

The flow behind a BFS is complex and involves various unsteady features. A standard representation of the flow topology is provided in Figure 5.1. The flow wake can be divided into three main regions: the shear layer region, the reattachment zone and the relaxation region [48]. A BFS flow begins with an unsteady phenomenon that shows up behind the separation point and in the shear layer region, which arises from the roll-up of the vorticity sheet emanating from the separation point because of the velocity discontinuity in the Y direction [79]. The initial small scale vortices are created due to the Kelvin-Helmholtz instability (K-H mode). Further downstream, these small vortices merge together or amalgamate into larger vortices and create larger structures. This behavior is referred to as vortex pairing. As the flow progresses downstream, the process of rolling and the amalgamation of vortices continue and larger and larger scale structures are created until they grow to the order of the step height and the process is inhibited by the lower wall. Since the turbulent structures grow as they move forward in the shear layer, the high frequency K-H instability changes to lower frequencies and ends up with a mode, called step mode, which corresponds to the lowest frequency of the vortex pairing.

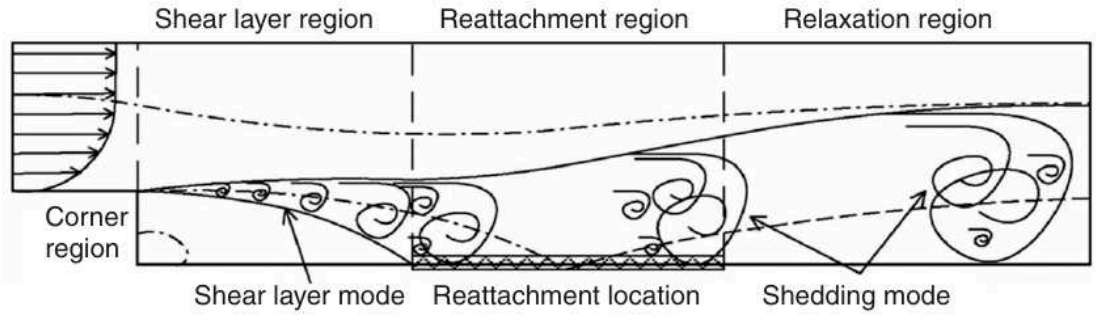


Figure 5.1 – Flow characteristics behind a BFS [48]

The shear layer structures, after attaining their maximum possible length scale, the step height, and being impinged on the wall in the reattachment region, slow down. This decreases their passing frequency and results in the gathering or amalgamation of the shear layer structures which increases their size (Figure 5.2). This happens in the reattachment region which is the location of the minimum value of the skin friction coefficient on the lower wall. The point at which the shear layer touches the wall, the reattachment point, is not fixed and moves upstream intermittently in bursts. Hasan [45] observed that this sudden upstream move splits the shear layer into two halves and compresses the fluid which is trapped underneath the shear layer. Most of the compressed fluid is ejected by pushing the shear layer outward and then the split shear layer remerges again. The downward and upward deflections of the shear layer represent a low-frequency flapping of the shear layer. Although the origin of the reattachment point movement is still controversial and several

other hypothesis were proposed by other researchers based on their observations [33, 44, 94], there is an agreement in the presence of an unsteady phenomenon, called flapping motion, which is a low-frequency instability involving the whole separated region. Finally, in the relaxation region, the large coherent structures shed downstream with no further amalgamation. This is referred to as the shedding mode.

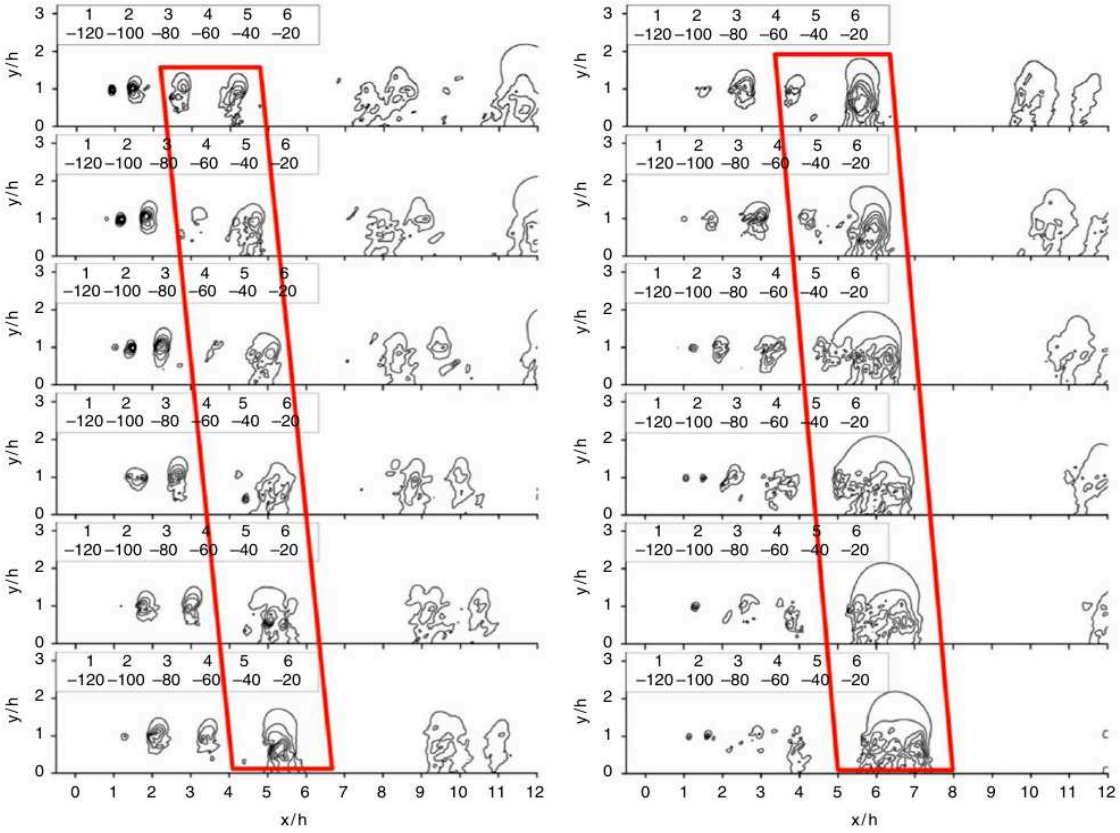


Figure 5.2 – Amalgamation in the reattachment region [48]

These modes and motions are described using the non-dimensional frequency (St). The Strouhal number is given by $St_h = fH/U_\infty$, where f is the mode frequency, H is the step height and U_∞ is the free stream velocity. In the study by Hu et al. [48] of a turbulent separation behind a backward-facing step with several point probes along the free shear layer, the frequencies corresponding to the shear layer rolling-up due to K-H instability, the vortex pairing and the flapping motion were found to be $St_{K-H} = 0.7$, $St_{pr} = 0.356$ and $St_{fl} = 0.006$ respectively. The frequency of the shedding mode which dominates the relaxation region was $St_{sd} = 0.074$.

The above short description of the unsteady features which exist in the the backward-facing step flow will be helpful for the better understanding of the analysis to come. In the following a backward-facing step case is simulated with several point probes in order to investigate the effects of averaging intervals in relation to the frequency of the flow structures and the unsteady features, on the averaged solution.

5.2 Simulation of the backward-facing step flow

The specific turbulent separated flow that is simulated here, is the one that was studied experimentally by Vogel and Eaton [105]. This particular flow was widely used during the

last decade to study the performance of hybrid turbulence models, particularly the IDDES model [41, 42, 86]. The objective of these studies was to model the attached boundary layer prior to the step and the boundary layer on the entire opposite wall in RANS mode, and after the flow separation, to have the model switch to LES mode. Also for the following simulation, the IDDES (see Section 2.2) is used as the turbulence model.

The computational domain based on the step height H (see Figure 5.3), runs from $-4H$ to $20H$ in streamwise direction ($x = 0$ corresponds to the step location). In the spanwise direction, the size of the domain is $4H$ and the channel expansion ratio is 5/4.

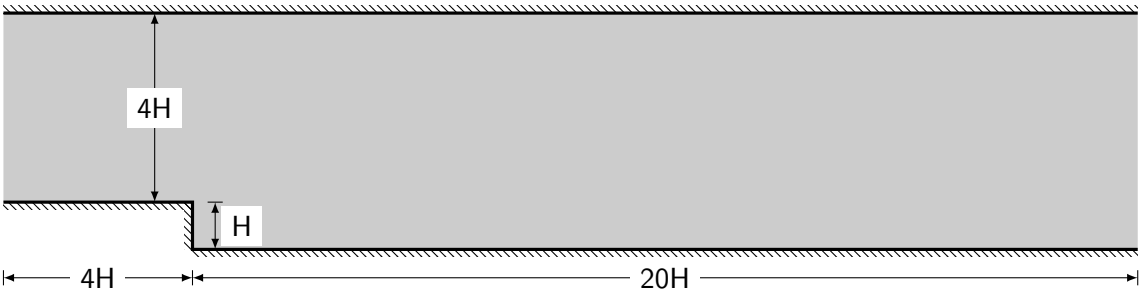


Figure 5.3 – Backward-facing step domain

Mesh generation and simulation parameters. For this simulation, adaptive refinement is not used. To activate the LES behavior of the IDDES model, a fine mesh which contains 9.2 million nodes in total is generated (Figure 5.4). The mesh settings are similar to the mesh used by Shur et al. [86]. It has appropriate clustering near the walls and in the vicinity of the step. Several refinement boxes are used to reduce the mesh size behind the step around the free shear layer area. The mesh is uniform in the spanwise direction (the number of cells is 80). The maximum grid sizes in streamwise and spanwise directions are equal to $0.1H$ and to $0.05H$ respectively, which corresponds to $\Delta x^+ = 120$ and to $\Delta z^+ = 60$ in wall units.

At the inlet, distributions of velocity and turbulence quantities are specified, which were obtained from a precursor RANS calculation conducted for the channel flow and correspond to the experimental value of the boundary layer thickness at that location. Virtual fluctuation inflow (Section 4.2.1) is not used at the inlet for this case, since the inlet fluctuations are dissipated in the upstream flow before reaching the separation point, by the use of the RANS turbulence model in this region. Periodic boundary conditions are specified in the spanwise direction. A no-slip condition is specified on the solid walls and on the outlet boundary a frozen pressure is applied.

The Reynolds number based on the step height, $H = 0.038\ m$, is equal to $28,000$ and the incoming turbulent flow boundary layer thickness is $1.07H$. A non-dimensional time step of 0.018 is chosen to ensure that the CFL number over the entire domain remains less than one, especially at the vicinity of the step, where the streamwise cell spacing is minimum.

Once the solution becomes independent of the initial conditions after 2500 time steps, the computation continues for 7500 time steps and the unsteady fields are averaged over the whole computation period for visualization purposes. In addition, the instantaneous velocity field, the pressure and the turbulence kinetic energy and frequency are averaged over each period of 100 time steps and saved in a series of files. These files will be used later on in Section 5.4 to perform the averaging over different periods.

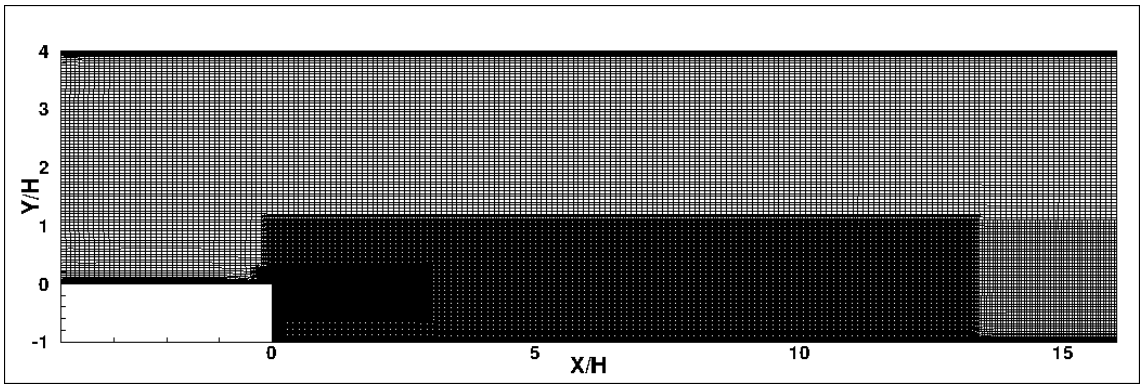


Figure 5.4 – Mesh resolution distribution over the backward-facing step domain

Comparison with experimental results. The time-averaged flow is compared with the experimental results of Vogel and Eaton [105]. First the averaged solution over the entire computation time is presented in Figure 5.5. In the averaged streamwise velocity, a separation zone is detected between the step wall, the lower wall and the free shear layer zone. According to the streamlines, the reattachment length (L_r), which is a critical parameter for the BFS flow, is around $6.4H$. It is comparable to the experimental data of Vogel and Eaton ($L_r = 6.66H$). A good agreement is also seen in the distribution of the mean friction coefficient along the lower and top walls (Figure 5.6). The under-predicted friction coefficient after $X/H = 11$ can be explained by the coarse mesh which was used at the end of the domain (Figure 5.4).

The physics of the flow and the evolution of the vortices can be seen in Figure 5.7. The vorticity sheet emanates from the separation point. With just one instant of the flow, it is not possible to see the interaction between the vortices. However, it can be seen that behind the separation point, the flow structures in the shear layer become larger, due to the rolling up and pairing of the small vortices and the amalgamation of the small vortices into the larger ones. In addition, the slow flow rate in the recirculation region and the reattachment point are also distinguishable by the velocity colored iso-surfaces of the second invariant.

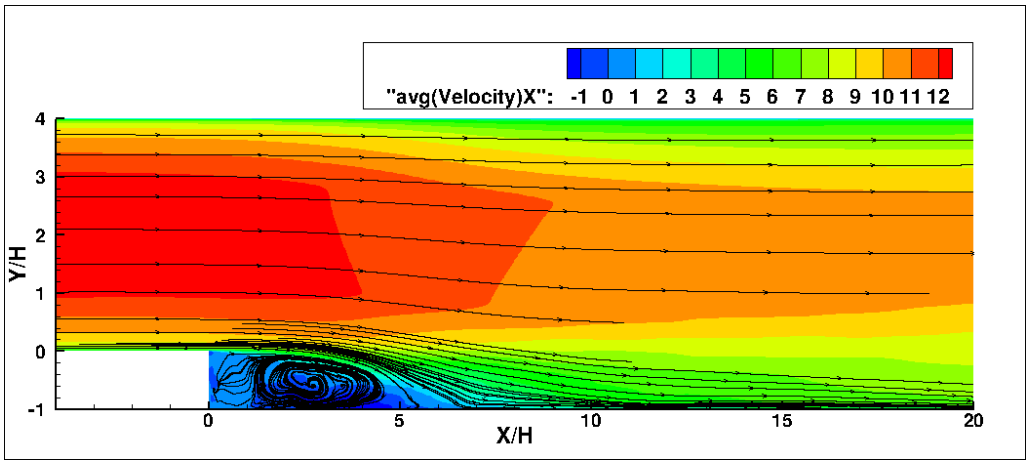
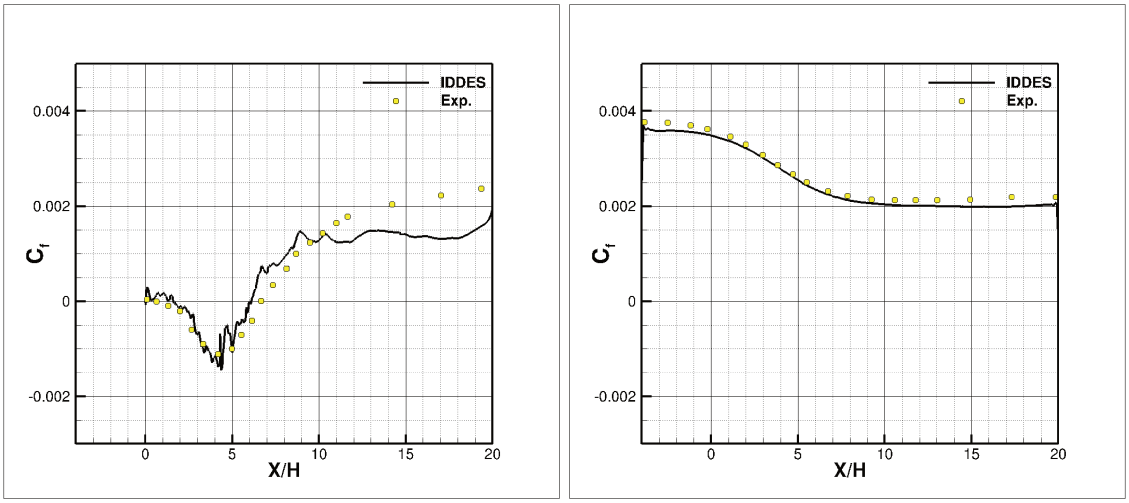


Figure 5.5 – Averaged streamwise velocity and streamlines

A comparison of the normalized averaged velocity and the normalized root mean square (RMS) velocity profiles predicted by the the simulations at several cuts with the experimental data, is presented in Figure 5.8. As seen in the figure, the velocity fields



(a) (b)

Figure 5.6 – Mean friction coefficient distributions along the (a): lower wall; (b): top wall

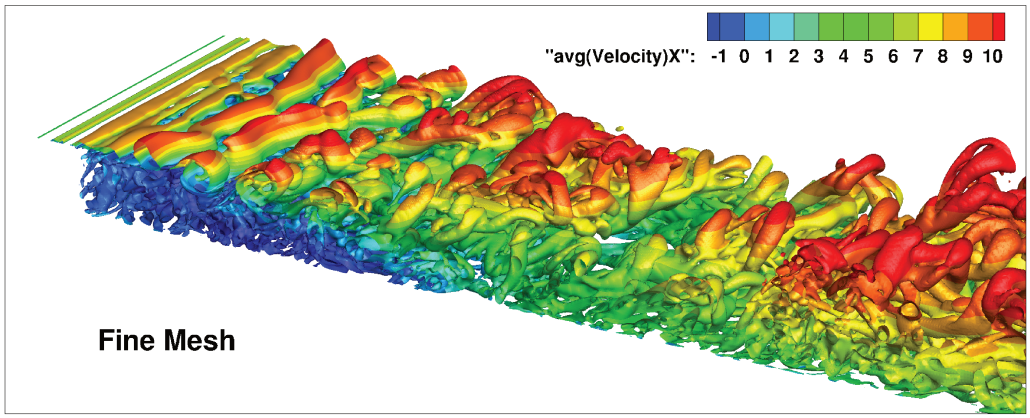


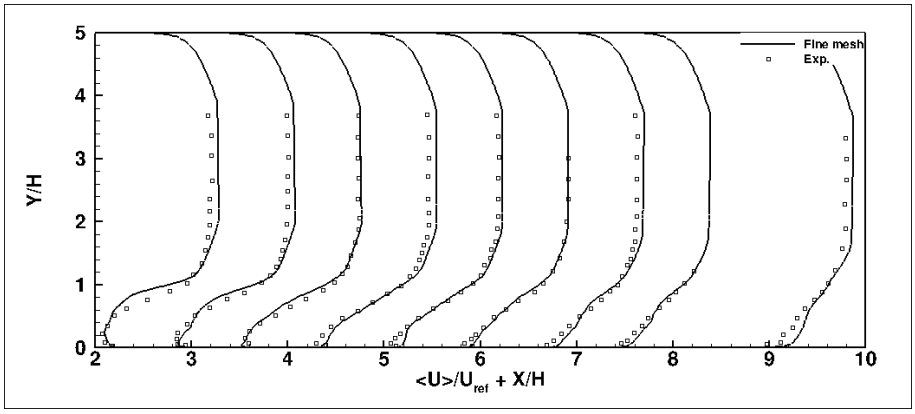
Figure 5.7 – Iso-surfaces of second invariant equal to 2500, colored with the mean axial velocity

agree well with the data. Thus, the IDDES model can be considered as adequate in both the attached flow region (ensuring a sufficient shielding of DDES from MSD) and the separation region (using the turbulence resolving capabilities of the model - WMLES), see Section 2.2.

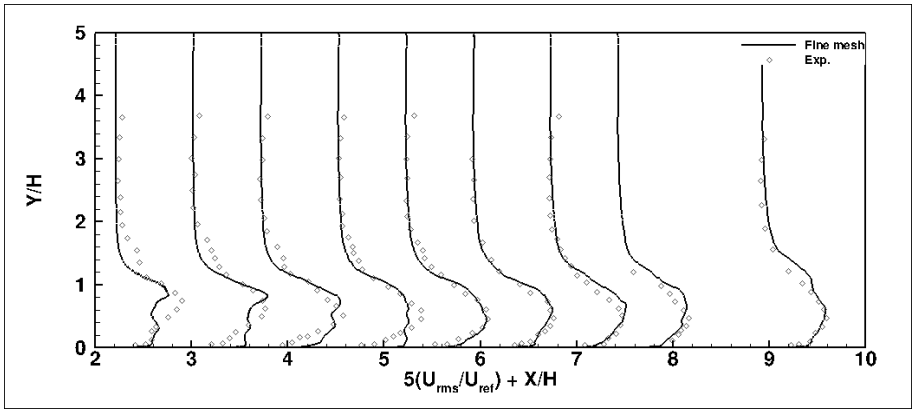
5.3 Analysis of point probe signals

For the BFS flow which is considered as a quasi-periodic motion, the right averaging interval is related to the frequencies of the unsteady flow features that are shown in Figure 5.7. To determine these frequencies, several velocity and pressure point probes are considered in different positions of the domain during the simulation (Figure 5.9). The locations of the points are selected in a way to have several points in the shear layer and reattachment and relaxation regions. The coordinates of the points are provided in Table 5.1.

To determine the frequency of the flow features, the fast Fourier transformation (FFT) is applied to the time signal in the point probes. In the shear layer region (Figure 5.10), $St_{pr} = 0.43$ which corresponds to the vortex pairing is seen at all the locations. In contrast, the frequency of the shear layer rolling-up due to K-H instability is only detected in the



(a) Normalized averaged velocity profile



(b) Normalized RMS velocity profile

Figure 5.8 – Comparison with experimental data. Profiles taken at $X/H = 2.2$, 3, 3.7, 4.5, 5.2, 5.9, 6.7, 7.4 and 8.9

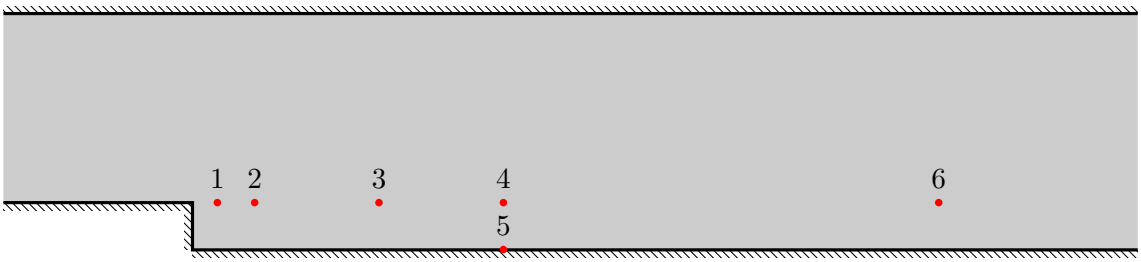


Figure 5.9 – Point probe positions

Table 5.1 – Coordinates of the point probes

Point	Region	x/H	y/H
1	Shear layer	0.52	1.0
2	Shear layer	1.32	1.0
3	Shear layer	4.0	1.0
4	Reattachment	6.67	1.0
5	Reattachment	6.67	0.0
6	Relaxation	15.8	1.0

Y-axis velocity. The high power density of the pairing frequency makes it difficult to recognize the K-H instability peak for the X-axis velocity. In addition another peak can be seen that is related to the low-frequency motion of the free shear layer called the flapping motion ($St_{fl} = 0.008$).

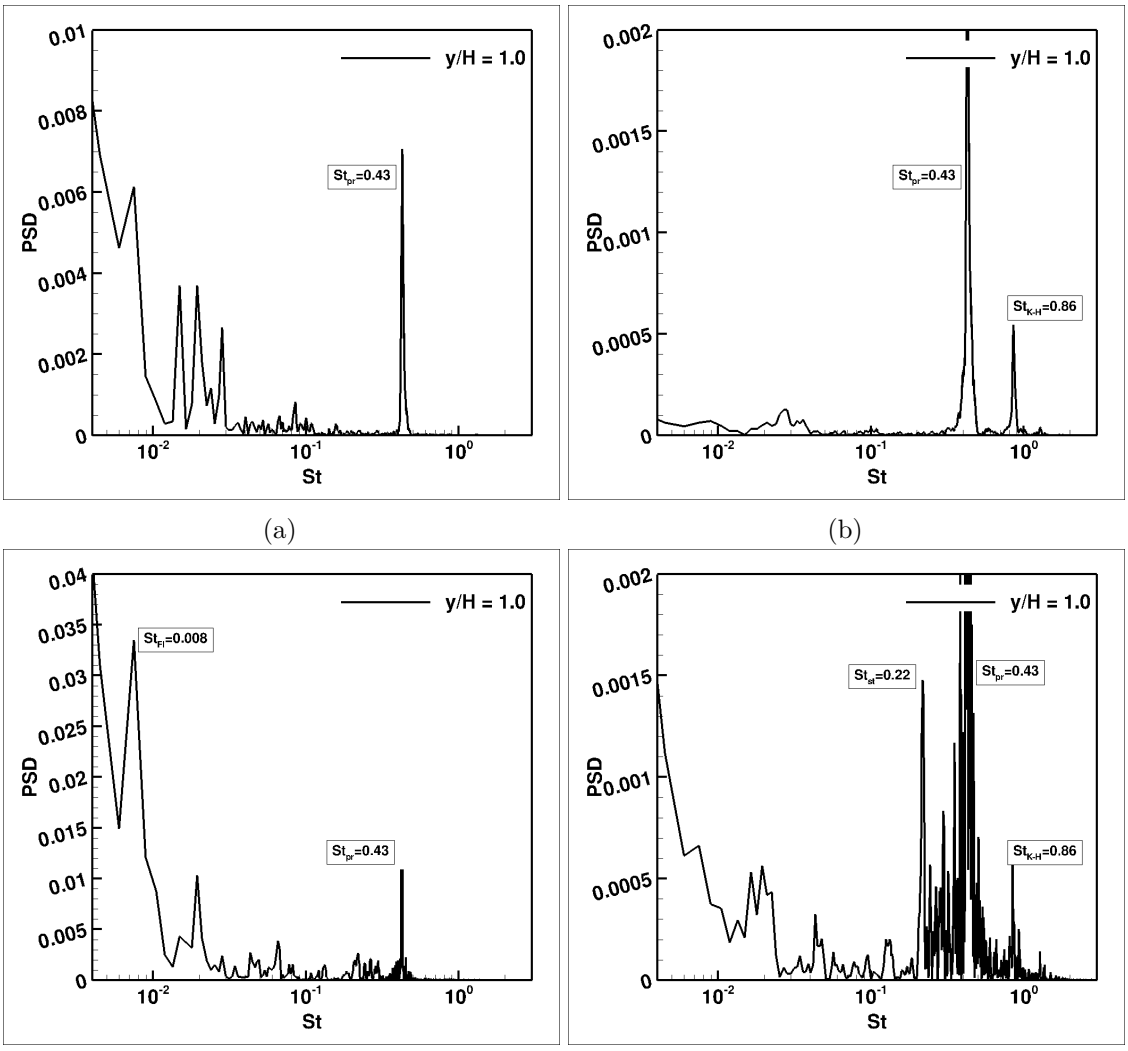


Figure 5.10 – Power spectra density of velocity in X (left) and Y (right) directions in the shear layer region; (a,b): Point 1 ($X/H = 0.52$); (c,d): Point 2 ($X/H = 1.32$).

Going downstream, the domination of the vortex pairing reduces and a new feature stands out (Figure 5.11): $St_{st} = 0.22$ which is related to the step mode. In the reattachment region and close to the reattachment point ($X/H = 6.67$), the step mode dominates and the pairing process is stopped. Here another peak is detected as well: $St_{sd} = 0.11$ corresponds to the shedding mode. Finally the effect of the flapping motion of the shear layer and the oscillation of the reattachment point is seen. At the end of the domain, in the relaxation region (Figures 5.11c and 5.11d) the dominating frequency is related only to shedding. The flapping motion is also detected in the relaxation zone with $St_{fl} = 0.008$. Among all the unsteady flow features, the lowest frequency corresponds to the flapping motion.

The same frequencies are seen for the pressure point probes (Figure 5.12). The vortex pairing as the major flow feature of the flow in the shear layer is replaced by the step mode in the areas between the shear layer and the reattachment regions. In the relaxation region,

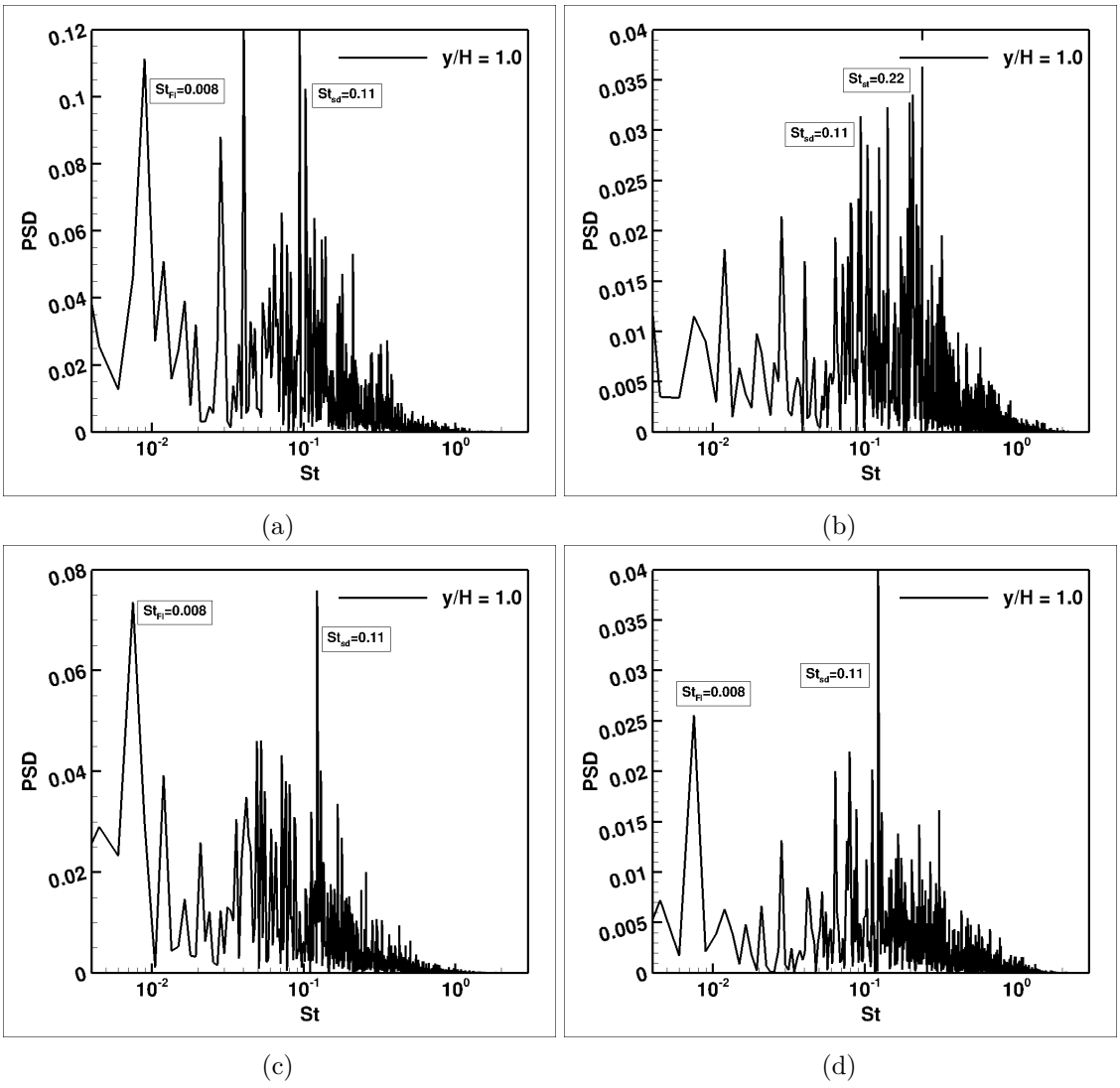


Figure 5.11 – Power spectra density of velocity in X (left) and Y (right) directions in the reattachment and relaxation regions; (a,b): Point 4 ($X/H = 6.67$); (c,d): Point 6 ($X/H = 15.8$).

the shedding mode dominates.

Knowing the dominating unsteady features in the flow and their frequencies helps in selecting an appropriate averaging interval. In the next section, based on this knowledge, different averaging intervals are selected and the corresponding averaged solutions are studied.

5.4 Analysis of different averaging intervals

When a refinement criterion is computed from the averaged flow, the rate of change in the averaged quantities will directly affect the behavior of the AGR. A short interval results in more changes in the averaged solution in terms of intensity and frequency, so the mesh will be modified frequently. In contrast, a longer averaging period leads to a mean solution with small fluctuations, therefore the mesh is not required to change as much. In other words, the variation of the averaged solution in time is a direct measure of the required

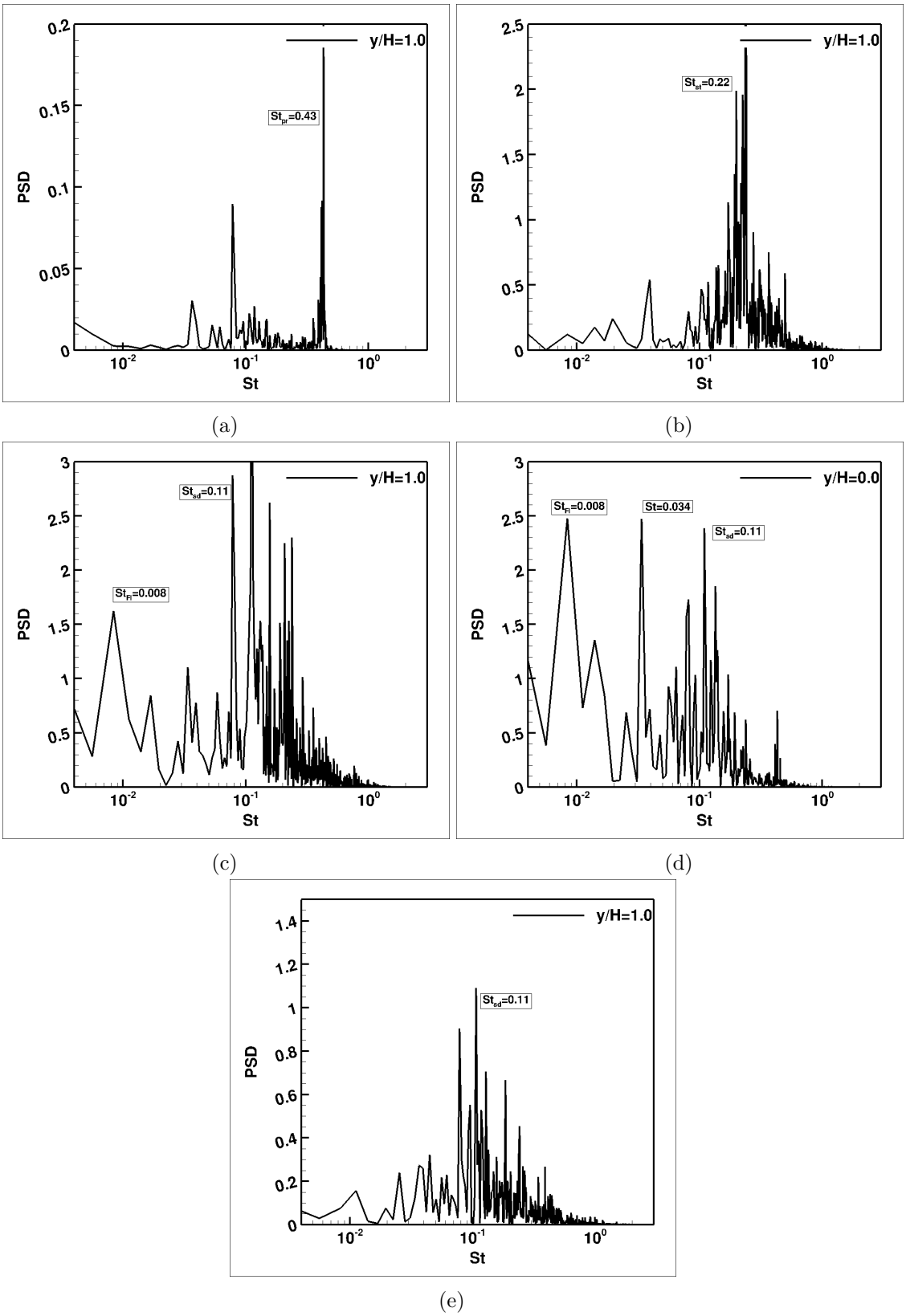


Figure 5.12 – Power spectra density of pressure at (a): Point 1 ($X/H = 0.52$) (b): Point 3 ($X/H = 4.0$) (c): Point 4 ($X/H = 6.67$) (d): Point 5 ($X/H = 6.67$) (e): Point 6 ($X/H = 15.8$).

mesh change. This requires the study of the effects of different averaging intervals on the averaged solution.

In Section 5.3, it was shown that in the flow behind a BFS, there are several frequencies which represent the behavior of the flow, but the two critical frequencies for the choice of the averaging window are related to the slow moving vortices of the shedding mode and the flapping motion. The corresponding non-dimensional frequencies of these low-frequency flow features are $St_{sd} = 0.11$ and $St_{fl} = 0.008$ respectively. The period length of the shedding mode is about 500 time steps and for the flapping motion, the period is 7000 time steps. These periods can be used as averaging interval to see how a change in the averaging interval affects the averaged solution.

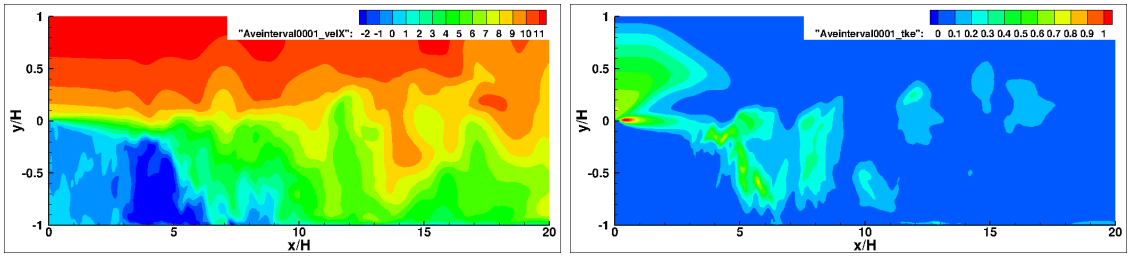
To investigate the influence of the averaging interval, the instantaneous solutions at each time step should be saved to perform the averaging as post processing, over different intervals. This requires a huge amount of disk space. In order to avoid that, instead of instantaneous solutions, the averaged solution over a small number of time steps is calculated and saved. For this specific case, after each 100 time steps, averaging was applied over the last 100 time steps and the averaged solutions were saved in a separate file. These files were then used to reconstruct averages over multiples of 100 time steps. As a consequence, the required memory for saving the data reduces by two orders of magnitude.

In Figure 5.13 the effects of several averaging intervals on the averaged streamwise velocity and averaged TKE are presented. For smaller intervals, changes in the averaged solutions are detected. Increasing in the size of the averaging interval makes these changes negligible, particularly for intervals longer than 5000 time steps. The effects of the high-frequency features in the shear layer region and also the shedding mode in the relaxation region, which are easily visible in Figure 5.13a, diminished significantly by increasing the averaging interval from 100 to 500 time steps. Between 500 and 1000 time steps, this reduction is more perceptible for flow features in the relaxation region than the ones in the shear layer region. It can be concluded that to reduce the effects of the lower-frequency flow features, i.e. shedding mode in the relaxation region compared to the high-frequency fluctuations in the shear layer, larger intervals are needed. Beyond 1000 time steps, which is twice the period length of the shedding mode, the changes in the averaged solutions continue but not as much as before. However, it is not easy to interpret precisely the effects of the averaging interval with this purely visual analysis of spatial fluctuations in the averaged solutions.

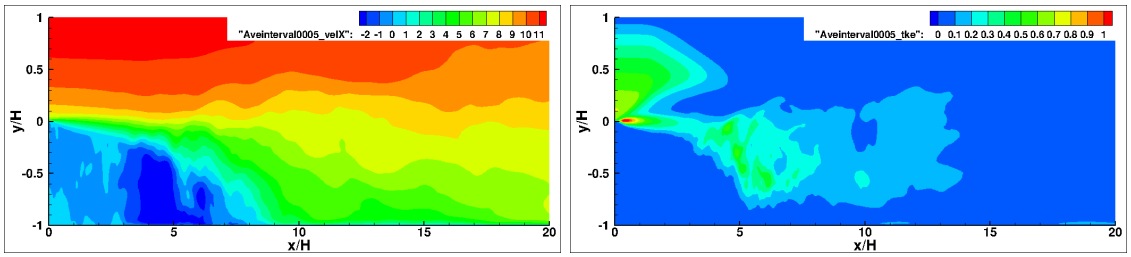
To have a more precise view and to see how the averaged solution evolves over time, the changes in the averaged solution with different time-averaging periods are presented in Figure 5.14 for several point probes of Table 5.1. For instance, for the interval with a size of 5000 time steps, the corresponding averaged velocity at time t is the mean of the last 5000 instantaneous velocities leading up to the time t.

For the averaging interval of 1000 time steps, large fluctuations can be detected in the averaged solutions. Increasing the averaging interval reduces the effects of the higher frequency fluctuations and limits the amplitude of the changes in the averaged velocity for a quasi-periodic flow like a BFS. It is implied that these increases will ultimately lead to the complete elimination of fluctuations from the mean solutions. This behavior is seen for all the point probes. It can be concluded that for chaotic behavior like the shedding, averaging over only 2 periods (1000 time steps) is not enough. To diminish its effects on the averaged solution, an averaging length of 10-15 periods (5000-8000 time steps) is needed. In contrast, the effect of the very low-frequency flow features like the flapping motion is still visible even for the longest intervals.

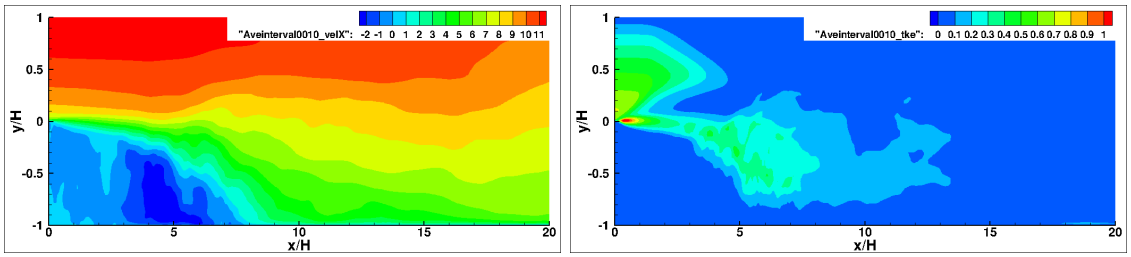
To avoid averaging over a very long interval for a periodic flow, it is possible to do the averaging over exactly one period. According to Figure 5.15, if the averaging is applied



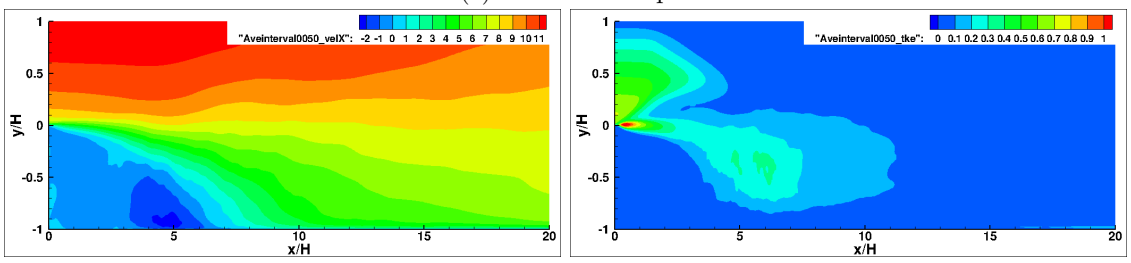
(a) 100 time steps



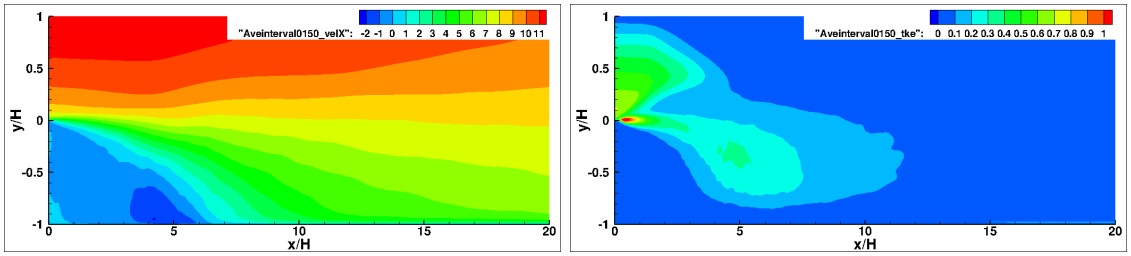
(b) 500 time steps



(c) 1000 time steps

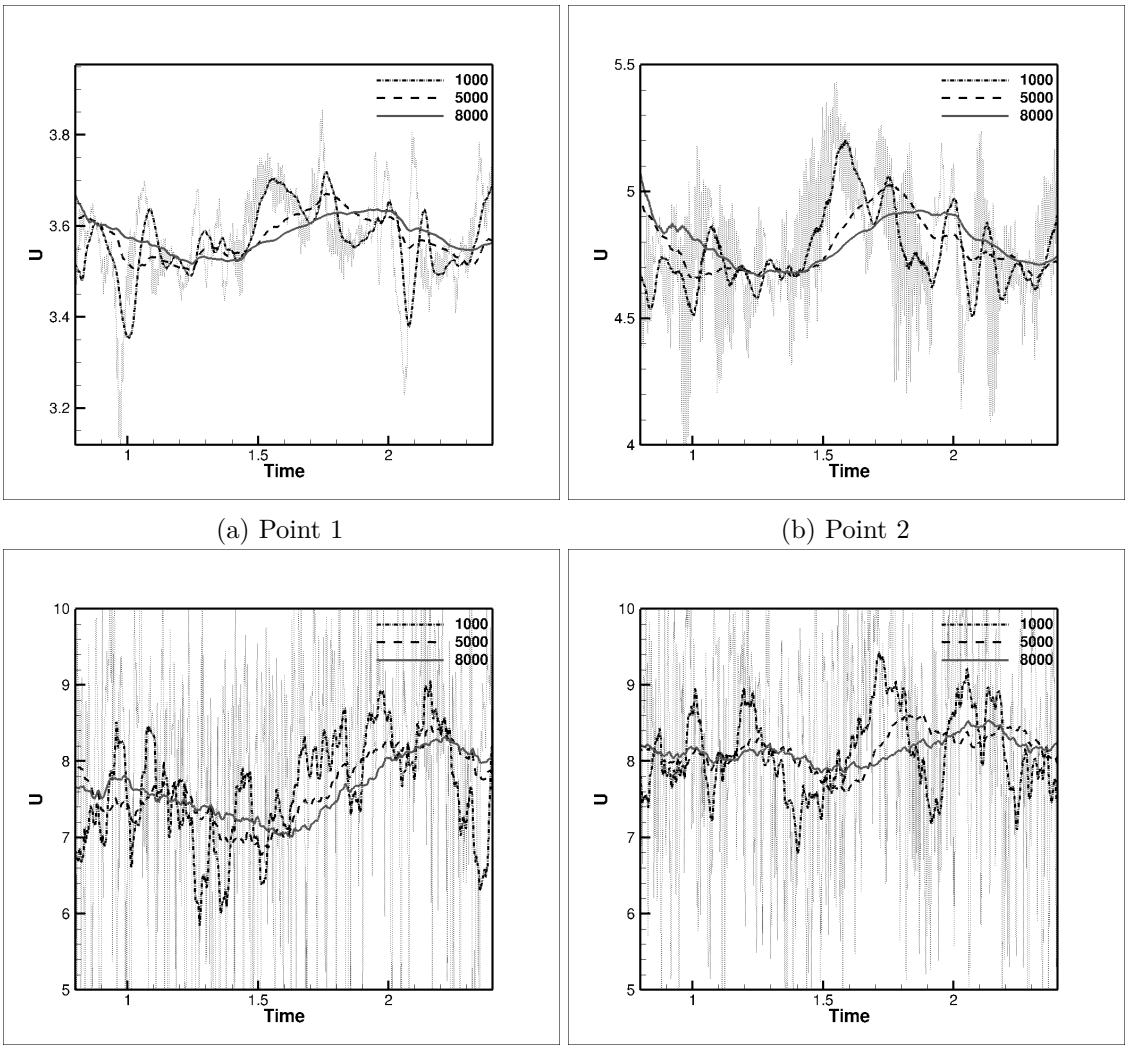


(d) 5000 time steps



(e) 15000 time steps (the whole computational time)

Figure 5.13 – Averaged streamwise velocity and turbulence kinetic energy, over several intervals



(a) Point 1

(b) Point 2

(c) Point 4

(d) Point 6

Figure 5.14 – Averaged streamwise velocity evolution in the points of Table 5.1, for several averaging intervals, The gray dotted line is the instantaneous signal.

over one period, no matter where the averaging begins, the result is constant. In addition, the results of the averaging over several periods and exactly one period are the same. However, for intervals longer (or shorter) than one period, the averaged value will not remain constant. Thus, for the BFS flow there might be an averaging period which removes the fluctuations of the high-frequency flow features and also gives an averaged solution close enough to a result with a very long averaging interval regarding the flapping motion. In other words, this interval would be a compromise between reducing the effects of flow features on the averaged solution and limiting the size of the averaging intervals which has a direct relation to the required memory.

To determine the optimal averaging window, corresponding to one period according to Figure 5.15, a new parameter is defined to show clearly the differences between the averaging intervals. The parameter which is used for comparison, is the summation of the absolute difference between two consecutive averaging intervals ΔT_1 and ΔT_2 with the same size (Figure 5.16).

$$\sum |\langle X \rangle_{i,\Delta T_1} - \langle X \rangle_{i,\Delta T_2}|$$

where $\langle X \rangle$ is the averaged local flow quantity.

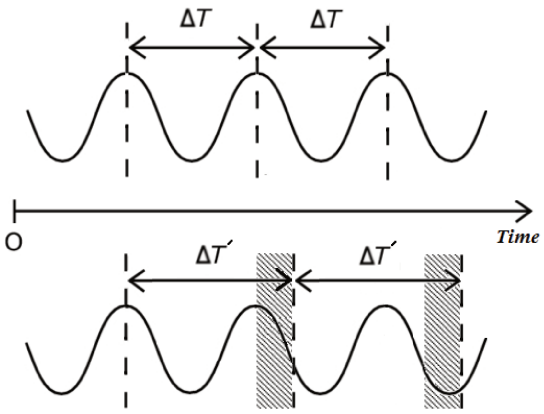


Figure 5.15 – Averaging over an interval larger than the signal period

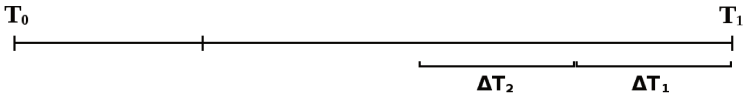
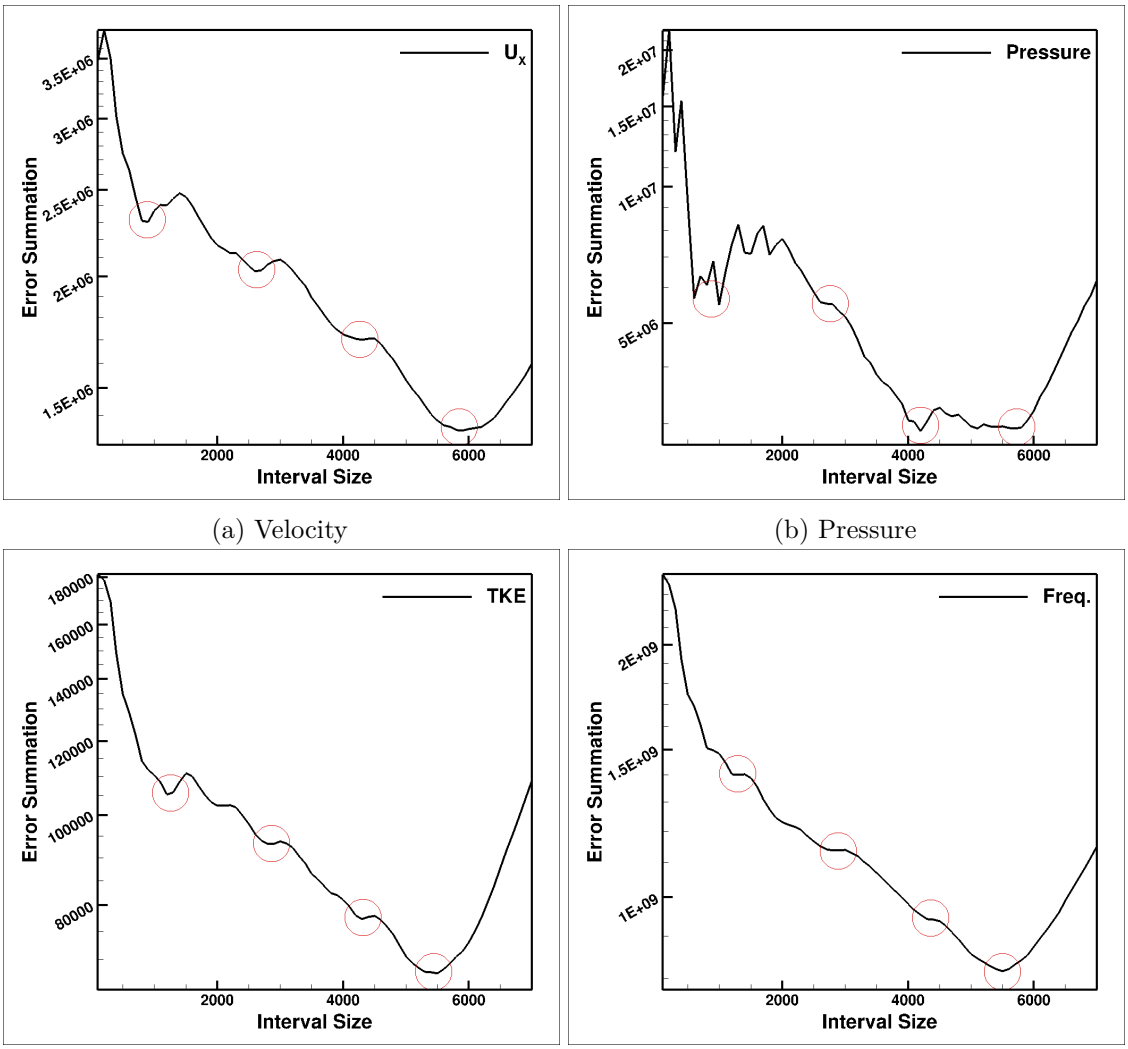


Figure 5.16 – Two consecutive interval with the same size

Figure 5.17 shows that larger intervals result in less difference between subsequent intervals for all the flow components, up to an interval of 5500 to 6000 time steps (the first circle from the right). For averaging intervals larger than that, the error summation increases. There are two reasons for this. First, the period of 6000 time steps is close to the length of the lowest detected frequency in the BFS flow: $St_{fl} = 0.008$ (7000 time steps) which corresponds to the flapping motion. Therefore, it can be considered as the characteristic period of this BFS flow. For the other intervals with larger or shorter size, the difference between two consecutive periods increases (Figure 5.15). The second reason is probably that the transient flow has an effect on the averaged solution for intervals longer than 6000 (ΔT_2), even though the simulation was intended to be long enough to remove the effects of the initial conditions before saving the solutions. Other circles in Figure 5.17 show the effects of averaging using intervals which match up with the frequencies of other features in the flow. For instance, for the intervals with lengths of 4200, 2800 and 1000 time steps, the related frequencies are $St = 0.013, 0.022$ and 0.04. These peaks can be seen in Figure 5.10 and Figure 5.11.

The distribution of the differences between two consecutive same-size averaging intervals is presented in Figure 5.18 for several intervals, for the streamwise velocity. In general, as the size of the averaging interval increases from 100 time steps, the large red-colored areas which correspond to the differences larger than 1 m/s, become smaller. Then, this reduction is replaced by an increase for the intervals larger than 6000 time steps. However, this behavior is not seen for all the regions of the flow. For instance, in the relaxation region the differences decrease, even as the interval size increases beyond 6000 time steps. For the larger intervals, the effects of the shedding mode which dominates the relaxation region are removed better. In the reattachment region, the differences decrease up to 6000 time steps but because of the large effects of the slow features of the flow in this region such as the flapping motion, the red areas are not removed and become larger for the intervals which are larger than 6000.

The local chaotic behavior of the flow makes it difficult to determine an exact period for the BFS flow, on which to perform the averaging. However, for an interval size which



(a) Velocity

(b) Pressure

(c) Turbulence kinetic energy

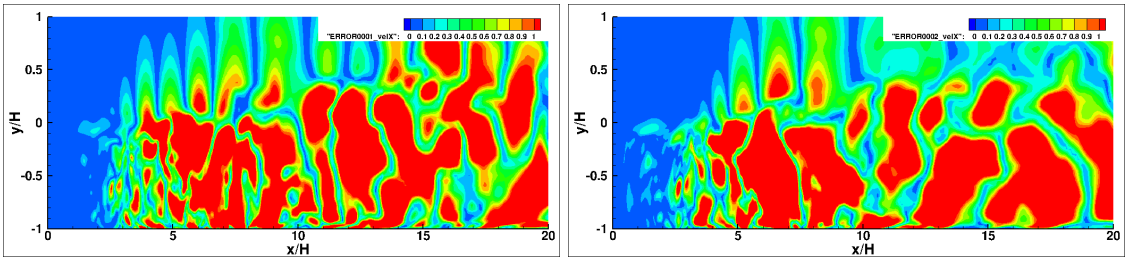
(d) Turbulent frequency

Figure 5.17 – Summation of the absolute difference of two consecutive intervals

is close to one period of the flapping motion, it is possible to diminish or even remove the effects of some features in the BFS flow such as shedding mode and also obtain an averaged solution which has the least difference with the averaged solutions of very long averaging intervals.

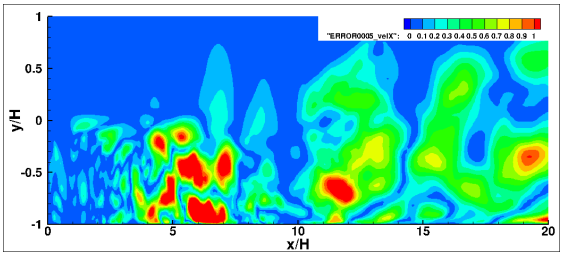
5.5 Conclusion

The mesh changes induced by adaptive grid refinement, that may perturb the turbulence and introduce interpolation errors, can be limited by basing the refinement criterion on time-averaged quantities. However, the size of the averaging interval has an effect on the intensity and the frequency of the fluctuations in the averaged solution. Therefore, it determines the behavior of the AGR and thus the adapted mesh. Time-averaging over the whole computational time is the easiest but not always the best choice. Instead, the averaging can be applied over a shorter sliding window. The selection of an effective interval size gives interval averaging an advantage over using the entire computational time. With a short averaging interval, the averaged solution is changed faster by the fluctuations

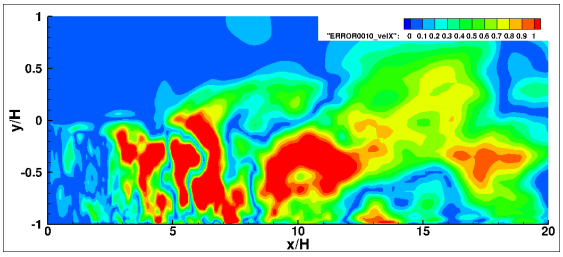


(a) 100 time steps

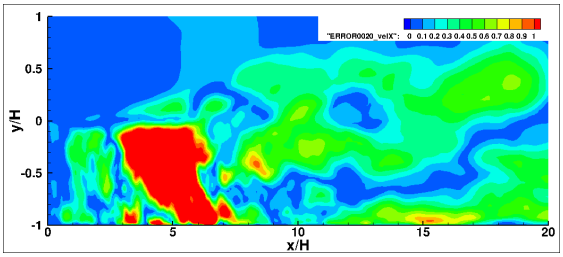
(b) 200 time steps



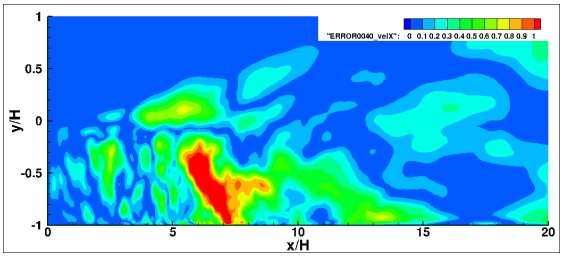
(c) 500 time steps



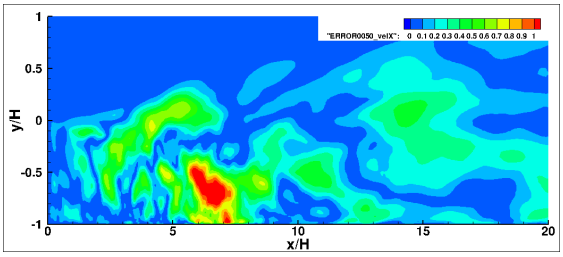
(d) 1000 time steps



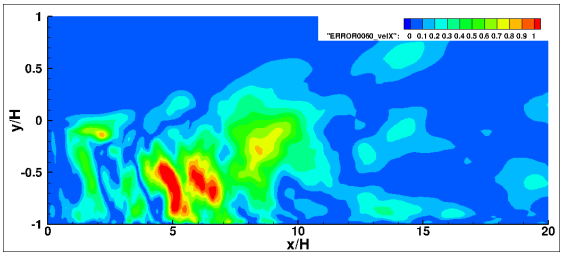
(e) 2000 time steps



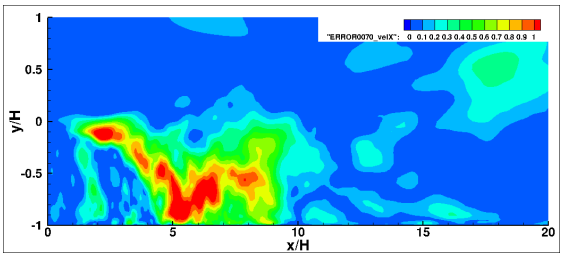
(f) 4000 time steps



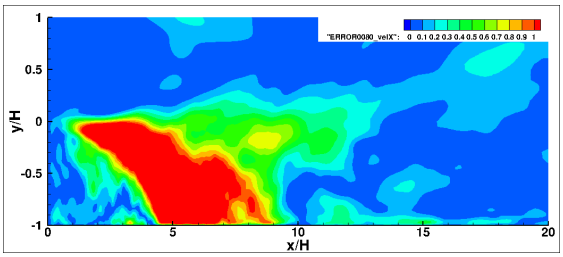
(g) 5000 time steps



(h) 6000 time steps



(i) 7000 time steps



(j) 8000 time steps

Figure 5.18 – Averaged streamwise velocity over several intervals

and thus AGR responds faster to large flow changes. However, finding this interval is not possible without considering the physics of the flow.

In this chapter, a turbulent flow behind a backward-facing step is studied. The frequency analysis of the unsteady flow features showed that there are several modes in the flow and that each of them is important in different regions of the flow. The vortex pairing in the shear layer region corresponds to the frequency of $St_{pr} = 0.43$. The step mode ($St_{st} = 0.22$) takes place at the end of the shear layer and the reattachment region. The shedding mode ($St_{sd} = 0.11$) in the relaxation region and the frequency of $St_{fl} = 0.008$ that is related to the flapping motion of the shear layer are considered as the two low-frequency characteristics of the BFS flow.

The averaging interval study of the BFS flow revealed that for short sliding averaging windows, the features of the flow show their effects on the averaged solution as high-amplitude and high-frequency signals. By an increase in the size of the interval to around 10 periods of the shedding frequency, it is possible to remove the effects of most features of the flow and reduce the fluctuations in the averaged solution. However, for the low-frequency features such as the flapping mode, an even longer averaging interval is needed. In that case, since the BFS flow is quasi-periodic, it is possible to find a characteristic signal for the entire flow and to apply the averaging over one complete period of it, instead of a very long interval.

To find the proper period length, the differences between two consecutive intervals with the same size were compared for several interval sizes. In different regions of the flow, an increase in the size of the averaging interval does not always lead to a decrease in the differences, since the dominating features and therefore the local signals are not the same. However, in a general view, based on the summation error of the entire domain (Figure 5.17) and the local differences between two consecutive intervals (Figure 5.18), it can be concluded that an averaging interval of about 1000 time steps (2 complete periods of the shedding mode) will be enough to diminish the effects of the high-frequency flow fluctuations in the shear layer and the shedding effects in the relaxation region on the averaged solution. However, a longer averaging interval is required to remove these fluctuation effects completely. The optimal size for an averaging interval is between 5000 to 6000 time steps which is 10-15 periods of the shedding mode and is close to the one period of the flapping motion. Averaging over this interval size removes the effects of the flow features like the shedding mode and it gives a relatively close approximation to the solution over a very long averaging interval. By averaging over this averaging interval, the changes in the averaged solution are reduced significantly. even though it is not possible to remove the effects of the slow moving flapping motion and the averaged solution will remain under the influence of this flow feature.

Selecting any of these averaging intervals will affect AGR behavior and lead to a different amount of refinement in the mesh. Therefore, depending on which unsteady flow features are most important, or to what extent the mesh changes are tolerated, a short or a long averaging interval can be chosen. What happens in all cases is, that the fluctuations in the solution are reduced to a desired extent by the averaging, and the adaptation based on this averaged solution follows the slower evolution of the flow.

Chapter 6

Analysis of average-based refinement criteria

As discussed in previous chapters, coupling adaptive grid refinement (AGR) and hybrid RANS/LES models will affect the performance of the turbulence model and also introduce extra truncation errors into the system. A possible solution, especially in configurations for which the spatial resolution requirements do not vary significantly over time (statistically steady or periodic turbulent flows), is to base the refinement criterion on the averaged flow (Section 5.1). This chapter discusses the development of such criteria.

Two approaches can be used to base AGR on the averaged solution: the first one is to perform the averaging of the instantaneous solutions over a proper averaging interval determined according to the flow physics (see Chapter 5). Then, before each refinement step, the refinement criterion is computed from this averaged solution. In the second approach, the refinement criterion is computed from the instantaneous solution at each time step and then the averaging is applied to the computed instantaneous criteria over the same averaging interval. There are several differences between these two approaches which will be discussed in detail in this chapter.

Selecting a proper averaging interval is important because, as discussed in Chapter 5, it will affect the rate of change in the averaged solution, which modifies the behavior of AGR. For instance, for the backward-facing step (BFS) case in the previous chapter, the effects of high-frequency flow features in the shear layer and the shedding mode in the relaxation region are averaged out when an interval of around 6000 time steps is used (10-15 periods of the shedding mode or almost one period of the flapping motion). Thus, the averaged solution varies little in these regions of the flow, so less grid variation will be expected and a more or less constant mesh will be obtained. In opposite, using a shorter interval like 1000 time steps (2 periods of the shedding mode) increases the variation of the averaged solution in the relaxation region due to the shedding mode. Therefore, more and faster variation will be applied to the mesh by the AGR in this region. Hence, the effects of the averaging interval on the average solutions lead to behavioral changes in the AGR that should be considered.

The effect of small-scale eddies in the regions of the flow where the velocity is low, like the recirculation zone for the BFS, is also studied. The small eddies in the recirculation zone may be transferred to its upper shear layer by the circulation in the region and affect the destabilization of the shear layer. However, averaging removes the small fluctuations in the local flow and since the AGR is based on the averaged quantities, this may reduce the amount of the refinement in these regions and therefore dissipate the local turbulence. Since this is a potential problem for any average-based refinement, the likely effects of these small eddies are investigated in this chapter.

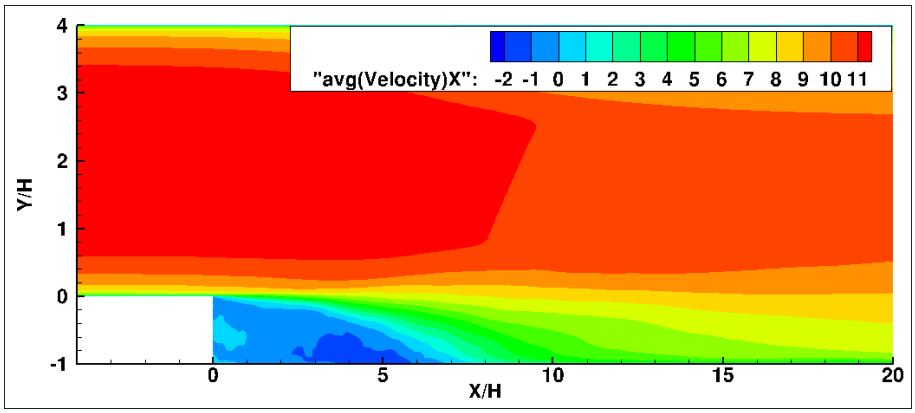
To study the effects of the averaging interval and the two averaging approaches on the behavior of AGR, the already described BFS case is considered (see Section 5.2 for more details). In Chapter 5 the simulation was performed using periodic boundary conditions in the spanwise direction. Since it is currently not possible to use AGR and periodic boundary conditions together in the ISIS-CFD solver, the same simulation as the one described in Section 5.2 is first repeated with mirror boundary conditions in the spanwise direction instead of the periodic boundary conditions. The results of this reference simulation are presented in Section 6.1. Another reference simulation in Section 6.2 shows the frequent mesh changes when AGR is based on the instantaneous solutions without performing any averaging. The computation starts with an initial coarse mesh and AGR is used to follow most of the turbulent structures without limitation. In Section 6.3, the two averaging approaches defined above are described in detail and their effects on the behavior of the AGR and the final solution are studied. The size of the averaging interval as an influential parameter on the performance of the AGR is investigated in Section 6.4. The importance of the small eddies in the recirculation zone is assessed in Section 6.5 by adding a refinement box to the initial mesh over this zone. The chapter ends with conclusions.

6.1 Periodic and mirror boundary conditions

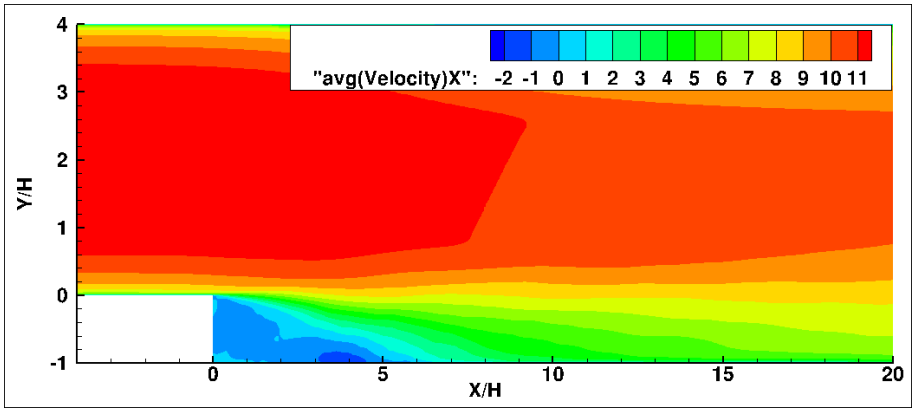
In Section 5.2, a turbulent flow behind a backward-facing step is simulated using the IDDES turbulence model on a high-resolution mesh without adaptive grid refinement. Periodic boundary conditions are applied in the spanwise direction. Since in the ISIS-CFD solver, the periodic boundary condition for AGR is not yet defined, the mirror boundary condition will be used instead for all the computations with AGR in this chapter. As a reference for these computations, the fine-grid simulation from Section 5.2 is repeated with mirror boundary conditions. Figures 6.1-6.5 present the effect of changing the spanwise boundary condition.

Using mirror boundary conditions results in a larger recirculation zone (Figure 6.1) and pushes the reattachment point toward the downstream direction ($X/H = 6.8$), compared to $X/H = 6.5$ for the simulation with periodic boundary conditions and $X/H = 6.6$ for the experimental data. The same thing happens for the resolved turbulence: as presented in Figure 6.2, the peak of the resolved turbulence kinetic energy moves downstream. The mean friction coefficient is also affected by the change in the spanwise boundary conditions (Figure 6.3). The solution along the lower wall moves to the right compared to the results of the simulation with spanwise periodic boundary conditions and the experimental data. However, after the reattachment and particularly at the beginning of the relaxation region, the results are in good agreement with the experiment. Right after the refinement box ($X/H = 13$), the jump in the friction coefficient is due to the abrupt coarsening of the mesh.

The average streamwise velocity and the average resolved turbulence kinetic energy for three X-cuts are presented in Figures 6.5 and 6.5. Each cut corresponds to a specific region of the flow: $X/H = 2.2$ lies in the shear layer, $X/H = 5.2$ in the reattachment region, and $X/H = 12$ in the relaxation region. The boundary layer thickness near the spanwise boundaries of the domain ($Z/H = \pm 2$) in the relaxation zone is significantly affected by the change in the boundary conditions (Figure 6.4). The same influence is seen in Figure 6.5 for the resolved turbulence kinetic energy. The mirror boundary conditions make a slow layer in spanwise direction, which causes some blocking. This blocking may accelerate the flow in the recirculation region and thus lengthen this region. Comparing the resolved turbulence in the shear layer region ($X/H = 2.2$) for both boundary conditions,



(a) Mirror boundary condition



(b) Periodic boundary condition

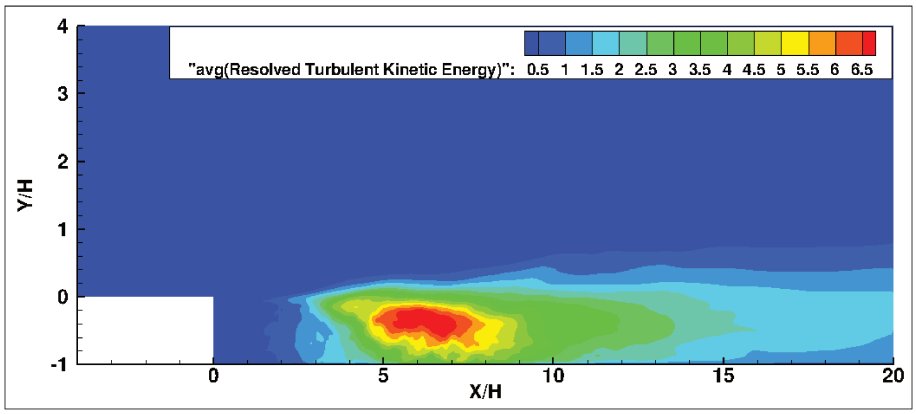
Figure 6.1 – Average streamwise velocity in the center plane

the delay in the generation of the resolved content for the simulation with mirror boundary conditions for the entire domain along the Z-axis is seen.

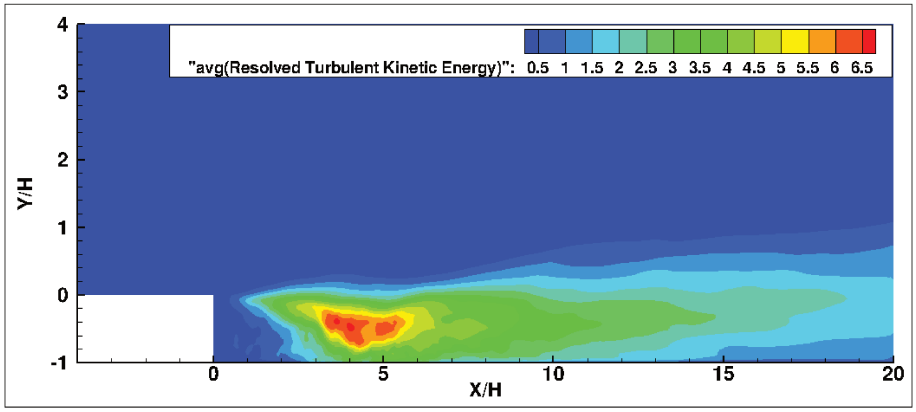
For the computations with AGR, mirror boundary conditions will be used as the spanwise boundary conditions. However, to reduce some of the effects of the spanwise boundary conditions, since the size of the domain in Z direction is large enough ($4H$), it is possible to only consider the solutions of a slice at $Z/H = 0$. But like this reference simulation with mirror boundary conditions, in computations with AGR it is expected to see the same differences in the reattachment length and the mean friction coefficient, compared to the simulation with the periodic boundary conditions and the experimental results.

6.2 Simulation using non-averaged AGR

Mesh and computation settings. Unlike the previous simulations, the initial mesh for a simulation with AGR should not necessarily be fine, since the adaptive refinement will create the fine mesh. For the BFS case, a simple unstructured coarse mesh is used (Figure 6.6). The grid sizes in all directions are twice larger than the biggest cell size of the previous simulations without AGR. A small refinement box is used close to the step tip to avoid large skewed boundary layer cells and to ensure a regular refined mesh near the separation point. The properties of the initial coarse mesh are presented in Table 6.1 and compared with the reference fine mesh which was used in the previous section.

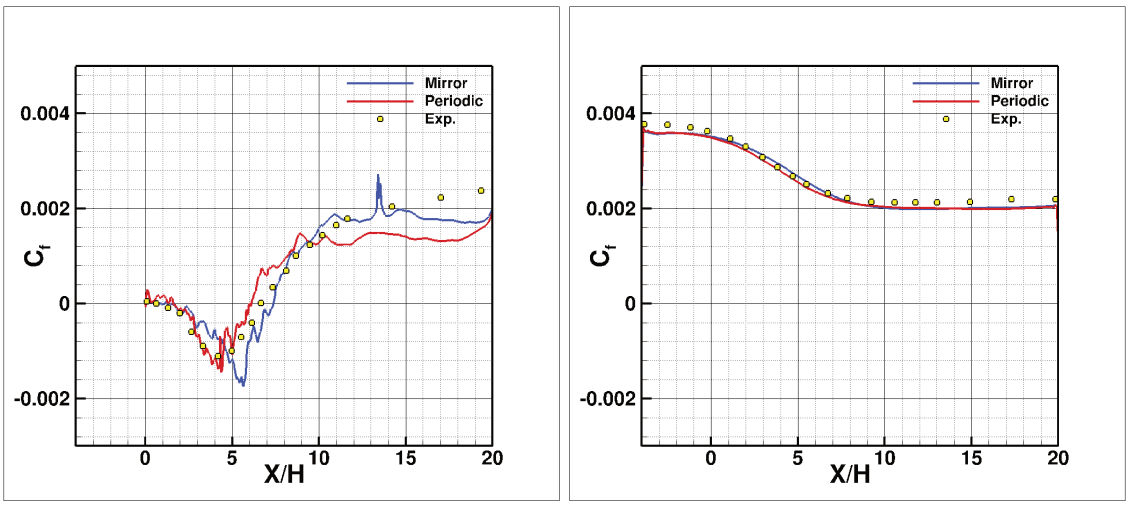


(a) Mirror boundary condition



(b) Periodic boundary condition

Figure 6.2 – Average resolved turbulence kinetic energy in the center plane



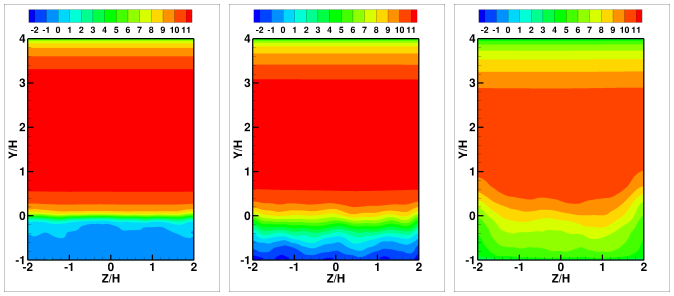
(a) Lower wall

(b) Top wall

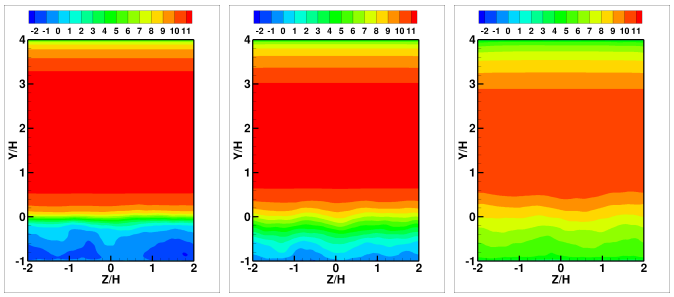
Figure 6.3 – Mean friction coefficient distributions

Table 6.1 – Properties of the initial mesh

Mesh name	$N_x \times N_y \times N_z$	Δx^+_{max}	Δy^+_{max}	Δz^+	N_{cells}
Coarse	$120 \times 50 \times 40$	240	120	120	480 880
Ref. Fine	$240 \times 100 \times 80$	120	60	60	9 283 200

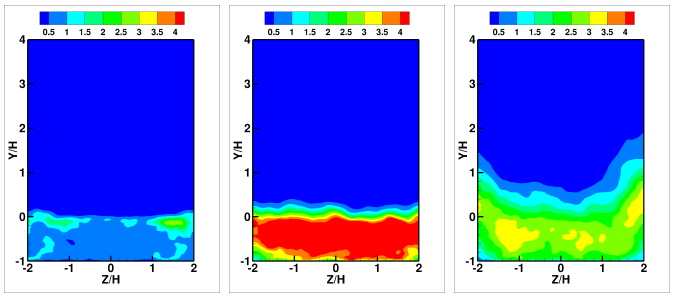


(a) Mirror boundary condition

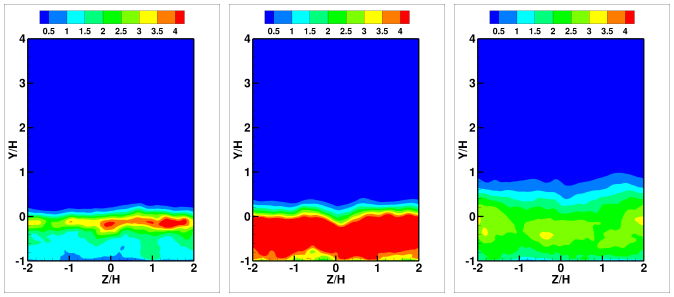


(b) Periodic boundary condition

Figure 6.4 – Average streamwise velocity with two different boundary conditions in spanwise direction at X/H = 2.2, 5.2, 12



(a) Mirror boundary condition



(b) Periodic boundary condition

Figure 6.5 – Average resolved turbulence kinetic energy with two different boundary conditions in spanwise direction at X/H = 2.2, 5.2, 12

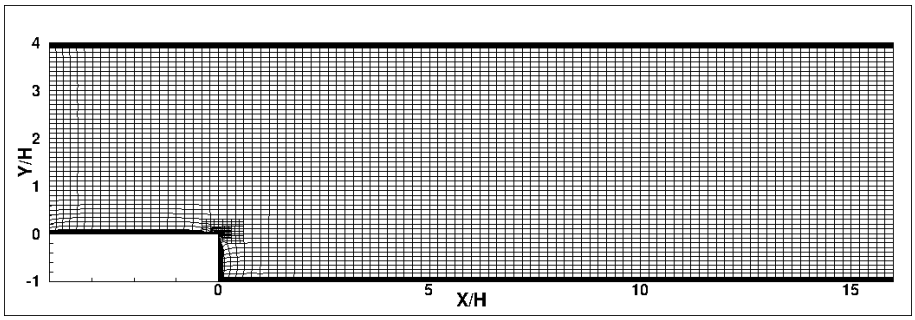


Figure 6.6 – Initial coarse mesh for the computations with AGR

The grid refinement starts once the solution becomes independent of the initial conditions (after 2500 time steps). The AGR parameters (Section 3.3) are set as follows: to follow all the small structures of the flow, the refinement process is repeated every two time steps. The second derivative of the flux components is used as the refinement criterion (see Section 3.3.2), which is computed based on the instantaneous flow field. The minimum cell size is set to the smallest grid size of the reference fine mesh in the previous section to allow the AGR to refine as much as the previous simulation. By trial and error, a threshold of $T_r = 0.01$ is set for the refinement. And finally, to avoid excessive computational cost, the refinement will be restricted in all directions for $X/H > 15$. Once the refinement is activated, the computation continues for about 7500 time steps which is close to one period of the flapping motion (see Section 5.1) in the BFS flow. This simulation is named *AGR-100* and its properties are presented in Table 6.2.

Non-averaged AGR results. In Figure 6.7, the variation of the number of cells during the computation is presented. The beginning of the refinement at $T = 0.15$ is associated with a sharp increase in the number of cells from half a million up to about 20 millions cells. This usually happens in computations with AGR which start from a very coarse mesh. The mesh is changed rapidly by AGR which perturbs the flow. This perturbation of the flow leads to more changes and more refinement in the mesh. This trend continues to some extent and afterwards, the flow fluctuations are reduced and subsequently a reduction in the number of cells is seen. When the flow is established, the mesh varies between 4 and 8 million cells and follows the unsteady flow features. For several instants, Figure 6.8 shows that the adaptation refines the grids to follow the moving vortices. The refinement is applied around the shear layer and also in the reattachment and the relaxation regions where the mesh changes constantly according to the changes in flow features. In other parts of the domain, especially in the recirculation zone where the flow rate is low, no refinement is applied by AGR. The local coarse grids behind the step show that the recirculation region grows during the course of the simulation.

In Figures 6.9 and 6.10, the solution of the computation with AGR based on the instantaneous flow (*AGR-100*) is compared with the reference simulation on the fine mesh. For the regions where the derivatives of the flow fields are small, like the recirculation zone, the adaptation does not refine the mesh and the small energy-carrying scales of the flow in these regions which were captured in the reference simulation, are missing (Figure 6.9b). In the recirculation zone, the absence of these small eddies, which may have an effect on the destabilization of the shear layer, results in a more stable and longer shear layer. The difference between the length of the recirculation zone for the two simulations can be easily detected by the averaged streamwise velocity colored iso-surface Q criterion. For the computation *AGR-100*, the reattachment length is about $X/H = 8.34$, 25% larger than the

Table 6.2 – Properties of the performed simulations, CA: Criterion of Averaged flow field, AC: Average of the instantaneous criterion

Computation name	Averaging method	Interval (time step)	T_r
Fine (fixed mesh)	-	-	-
AGR-100	NO Averaging	-	0.01
CA@-50	Time-averaged flow field	all	0.005
CA@-25	Time-averaged flow field	all	0.0025
AC@-25	Time-averaged Criterion	all	0.0025
AC@-20	Time-averaged Criterion	all	0.0020
AC@-15	Time-averaged Criterion	all	0.0015
AC50-25	Time-averaged Criterion	1250	0.0025
AC01-25	Time-averaged Criterion	25	0.0025
AC@-25B1†	Time-averaged Criterion	all	0.0025
AC@-25B2†	Time-averaged Criterion	all	0.0025

†A refinement box is added to the initial mesh.

experimental data (Table 6.3). Another difference between the two solutions is the poor representation of the shear layer in the computation *AGR-100* which is due to the low-resolution mesh around the shear layer region. The refinement threshold could be reduced to improve the resolution of the mesh. However, since the grids in the reattachment and relaxation regions are very fine compared to the local grids around the shear layer, a lower threshold which means more refinement for the entire domain, would result in an excessive refinement in these regions and therefore it increases significantly the computational cost.

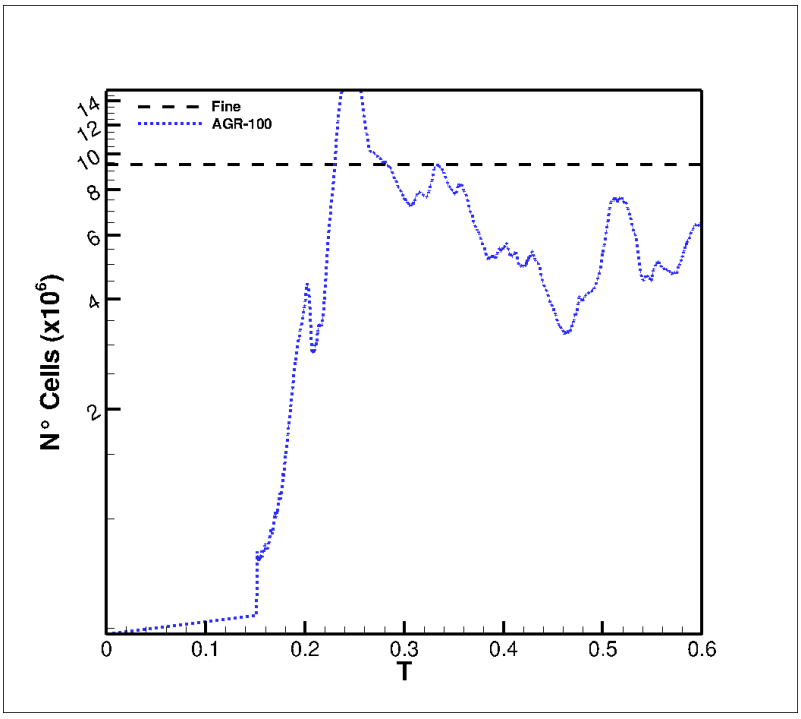


Figure 6.7 – Variation of the number of cells by AGR for *AGR-100*

The distribution of the mean friction coefficient along the lower and top wall is presented in Figure 6.10. There is a discrepancy with the experimental data along the lower wall, mostly for the reattachment and the relaxation regions where the mesh is constantly

Table 6.3 – Cell size and reattachment length

Computation	$N°$ Cells ($\times 10^6$)	L_r (X/H)	Error (%)
Exp.	-	6.66	-
Fine	9.3	7.3	9.6
AGR-100	6.6	8.35	25
CA@-50	0.7	9	35.4
CA@-25	2.4	7.14	7.2
AC@-25	1.3	7.9	18.7
AC@-20	2.5	7.0	5.1
AC@-15	9.9	6.9	3.6
AC50-25	2.5	7.7	15.6
AC@-25B1	2.0	6.5	-2.5
AC@-25B2	1.8	6.3	-5.4

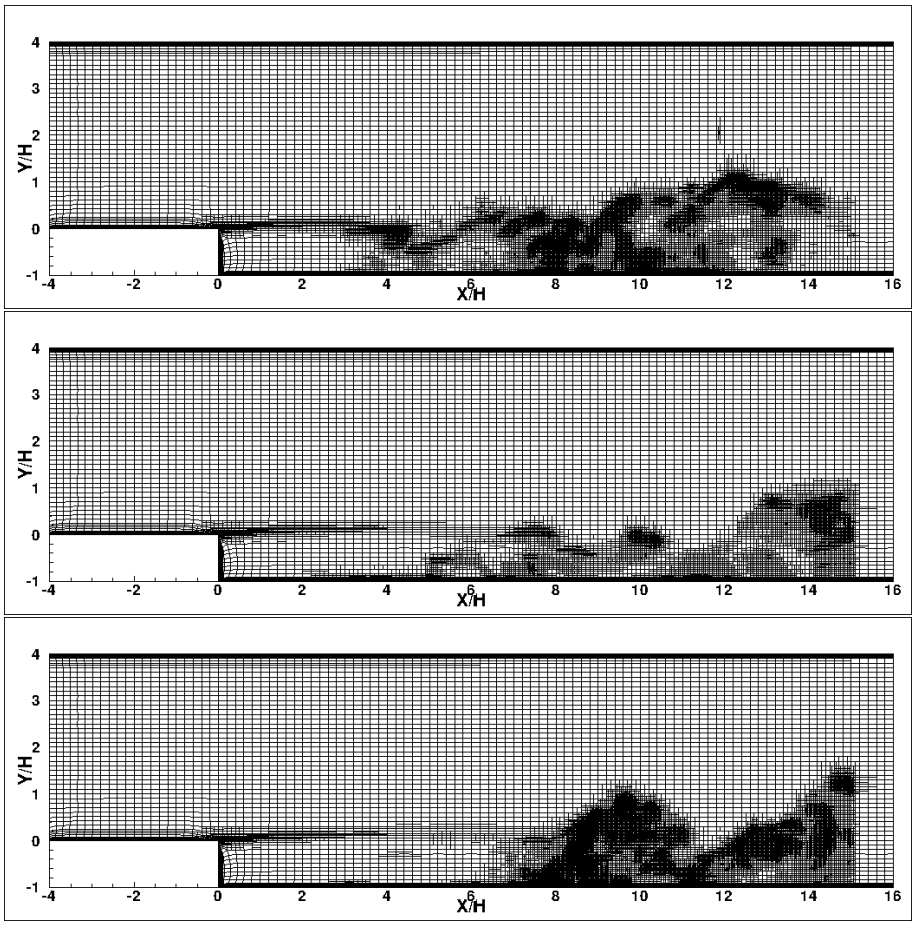
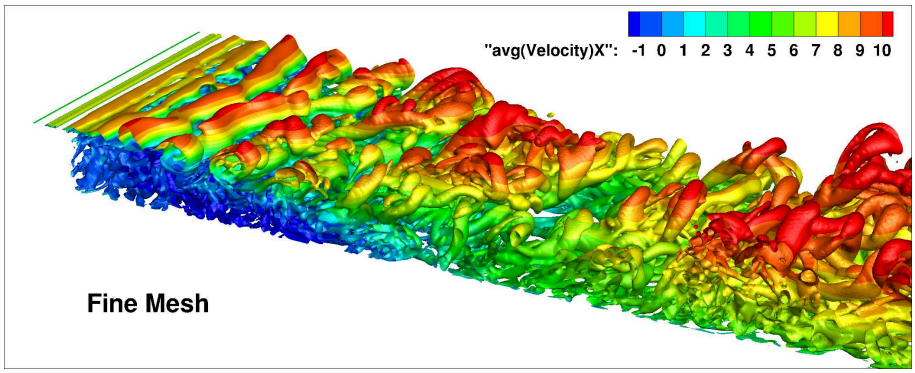
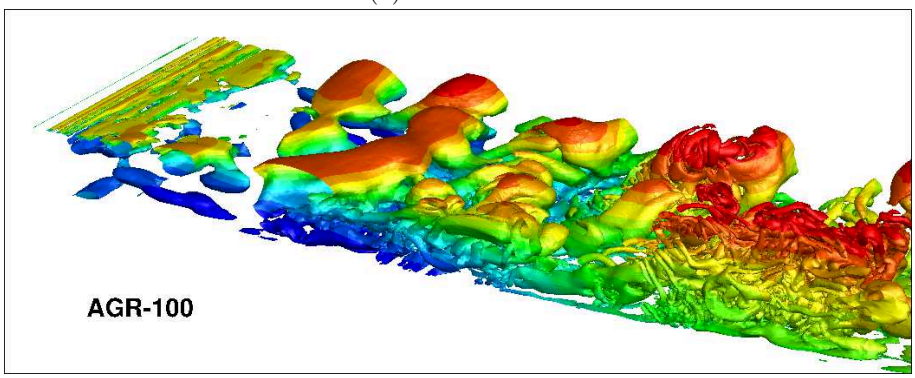


Figure 6.8 – Mesh evolution of *AGR-100* during the refinement process for the instants $T = 0.33, 0.42$ and 0.51

modified by AGR because of the changes in the flow structures. The likely explanation for this discrepancy is the interpolation errors in the transfer of the solutions between the old mesh and the new one, which perturbs the solution.

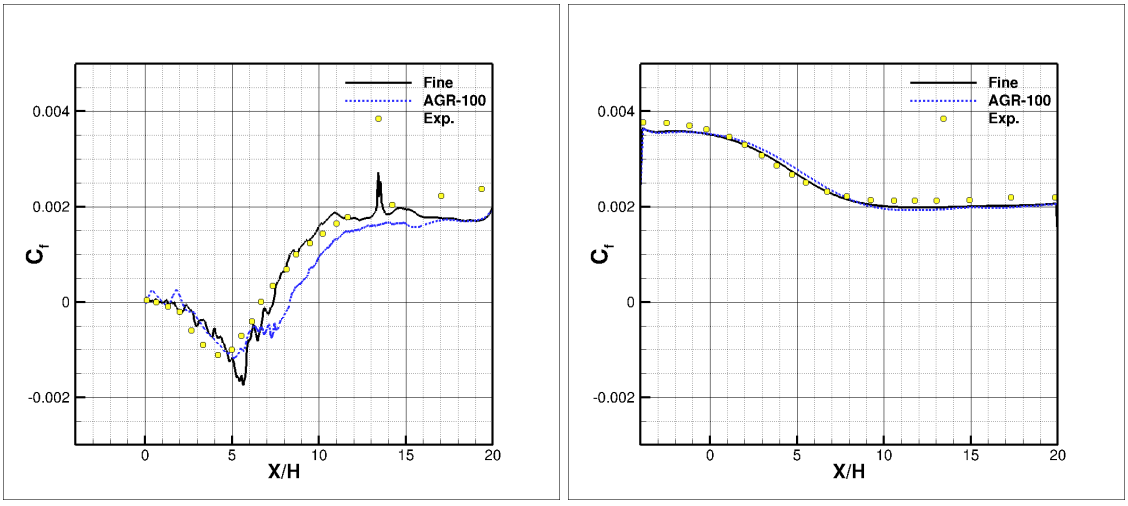


(a) Fixed mesh



(b) AGR-100

Figure 6.9 – Iso-surface of the second invariant $Q = 2500$ for the two reference simulations, colored by averaged streamwise velocity



(a) (b)

Figure 6.10 – Mean friction coefficient distributions along the (a): lower wall; (b): top wall

In conclusion, AGR based on the instantaneous solutions can follow the flow features by refining or derefining the mesh frequently. However, this capturing of the flow features

causes a significant variation in the number of cells during the computation which sometimes is undesirable, especially when the computational resources are limited. In addition, there is a large difference in the refinement criterion values between the recirculation region and the rest of the flow, for instantaneous DES flows. This difference leads to a low grid density in the recirculation zone compared to the rest of the domain. Reducing the threshold can increase the refinement in this region and helps in capturing the small local flow features. But since the threshold is a global parameter, AGR would increase the refinement of the entire domain, even in the locations where no further refinement is required, and the computational cost increases greatly. Finally, during each refinement the solution of the old mesh is interpolated to the new adapted mesh. This interpolation is associated with an error. Frequent refinement and large mesh changes at each refinement step increase the number of interpolations and thus the total error of the interpolations, which can perturb the final solution. The following sections investigate whether average-based grid refinement can solve these three problems.

6.3 Refinement based on averaged quantities

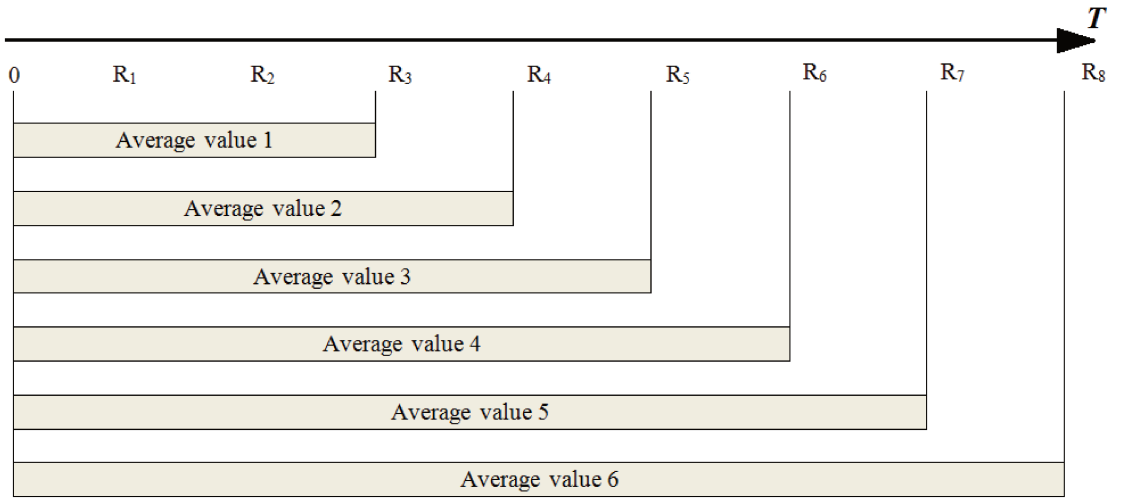
6.3.1 Averaging techniques

To limit the mesh changes during the adaptation process, AGR can be based on the averaged quantities, with less frequent refinement than in the previous test. Two different averaging approaches are studied in this section. In the first approach, the instantaneous flow fields (velocity and pressure fields) are averaged over a specific interval. Before each refinement step, the refinement criterion is computed based on these averaged fields. In the second approach, before each time step, the refinement criterion is computed from the instantaneous solution and averaging is performed over these instantaneous criteria. Then, in each refinement step, AGR refines or derefines the mesh based on the current average of the refinement criteria computed from the instantaneous solution. This approach is more expensive than the one in which the averaging is applied over the flow fields, in terms of computational cost and required memory. First, the refinement criterion is computed before each time step, instead of only before the refinement. And second, instead of 4 variables for the averaged solution (three for the velocities and one for the pressure), the 6 variables of the criterion tensor field (Section 3.3) must be saved for each set of averaging tables.

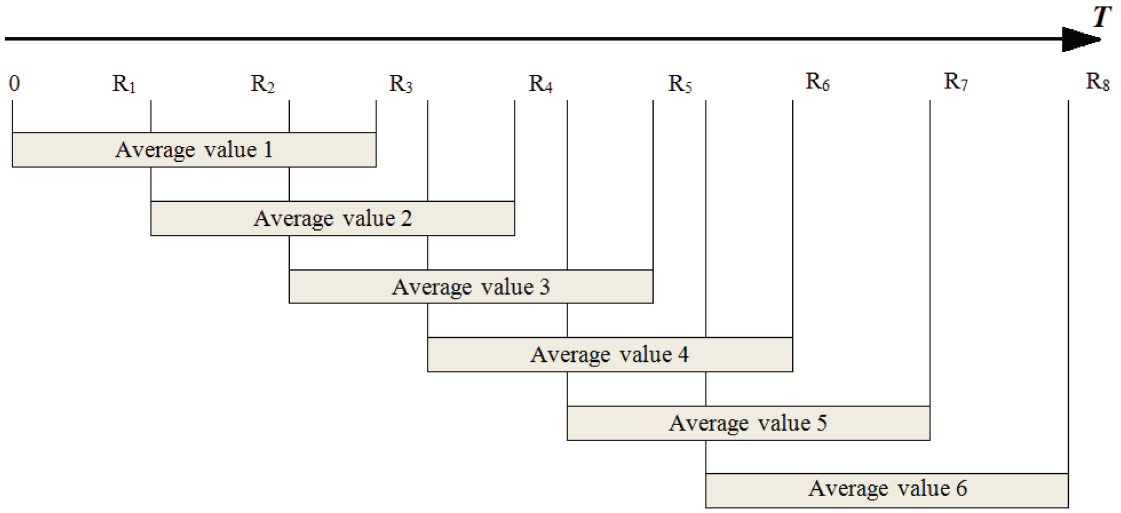
In addition, to perform the averaging over the solutions, two different averaging methods are proposed based on the size of the averaging interval. In the first method, the averaging is applied over the entire computational time, and in the second one a sliding averaging window is selected which slides through the computational time. Different implementations are required for each of these averaging methods. In Figure 6.11, the averaging schemes are presented.

For averaging over the entire computational time (Figure 6.11a), the instantaneous solutions in each time step are summed. The summation is saved in one set of tables which is updated before each time step. Before each refinement time step, the average value is computed by dividing the summation by the current number of time steps; this average is used to define the refinement criterion.

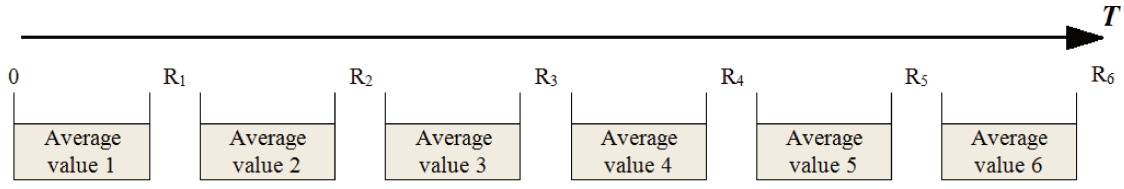
For a sliding window, in which the averaging is applied over a constant number of time steps, the averaging algorithm is more complex. The instantaneous solutions of each time step need to be saved separately to be used for the averaging. This is very expensive in terms of the required memory to save the solutions. Since the average solution is needed only at the refinement time steps, it is possible to use the same trick as in Section 5.4 to



(a)



(b)



(c)

Figure 6.11 – Averaging algorithms over (a): the whole computation time, (b): a specified number of time steps and (c): the time steps between the refinements; R denotes a time step where refinement takes place.

reduce the number of tables which should be saved. The average solution over intervals between refinement steps can be saved instead of saving the solution of each time step. In this case, two situations happen: if the sliding window equals the time steps between refinements, one set of tables is required to save the solution (Figure 6.11c) and the new average solution is overwritten in the existing set of tables after each refinement step. However, for longer sliding windows which must be a multiple of the interval between the refinements, a table will be required for each refinement interval in the sliding window (Figure 6.11b).

For the case which is used in this section, instead of every two time steps in the previous section, the refinement takes place every 25 time steps. When using a sliding window of 1250 time steps, after each group of 25 time steps, the averaged values are saved in one set of tables. For this interval which is a multiple of 25, 50 sets of tables are needed. After the first 1250 time steps, the averaged values of the next 25 time steps are saved in the tables which were associated with the oldest saved solution. For the refinement step, the averaging is applied over these 50 sets of tables and the averaged solution is used for the refinement. It should be noted that even 50 sets of tables is expensive since the standard ISIS-CFD uses 100-200 tables in total. So it means that the required memory increases by more than 150%.

These averaging methods are implemented in a dynamic library. This library is called by the ISIS-CFD solver before each time step. Depending on the averaging interval, either the entire computational time or a sliding window, one of the described algorithms will be used. The effect of the averaging interval is studied in Section 6.4.

6.3.2 Criterion of the averaged flow field (CA)

This section presents computations using the averaging approach where the criterion is computed from the averaged flow field. Like the previous computation with AGR in Section 6.2, the computation starts with a coarse mesh. The properties of the initial mesh are presented in Table 6.1. The computation runs for 2500 time steps to remove the transient effect of the flow, without any refinement. Then the averaging process starts and when the averaged solutions are computed over the first 25 time steps, the refinement is applied for the first time. For this simulation, the interval size is set to the entire computational time (7500 time steps max), so only one set of tables is enough to save the averaged values. With respect to the computation *AGR-100*, since the averaged quantities are used for the refinement, the magnitude of the criterion components is reduced. Thus, with the same threshold, according to Eq. 3.18 the requested grid size increases. To keep a similar number of cells, the threshold should be lowered. Therefore, two smaller thresholds are tested (0.005 and 0.0025).

To distinguish more easily between the different simulations, each computation is represented by a code. This code begins with two capital letters which corresponds to the averaging approach which is used in the simulation: «*CA*» indicates the Criterion based on the Averaged flow fields. The size of the sliding averaging window is specified by the number of tables used for saving the values, or by «@» which indicates the entire computational time. The last two digits on the right side of the code are related to the threshold. Thus, the two simulations considered here are called *CA@-50* and *CA@-25* (Table 6.2).

The results of these cases are presented in Figures 6.12-6.17. The variation of the number of cells shows an increase in the number of cells right after the AGR is applied for the first time (at time $T = 0.15$) in the computations *CA@-50* and *CA@-25* (Figure 6.12). This increase is far less than the computation *AGR-100*, even though the thresholds are lower for these two simulations. The final meshes of the two computations with different

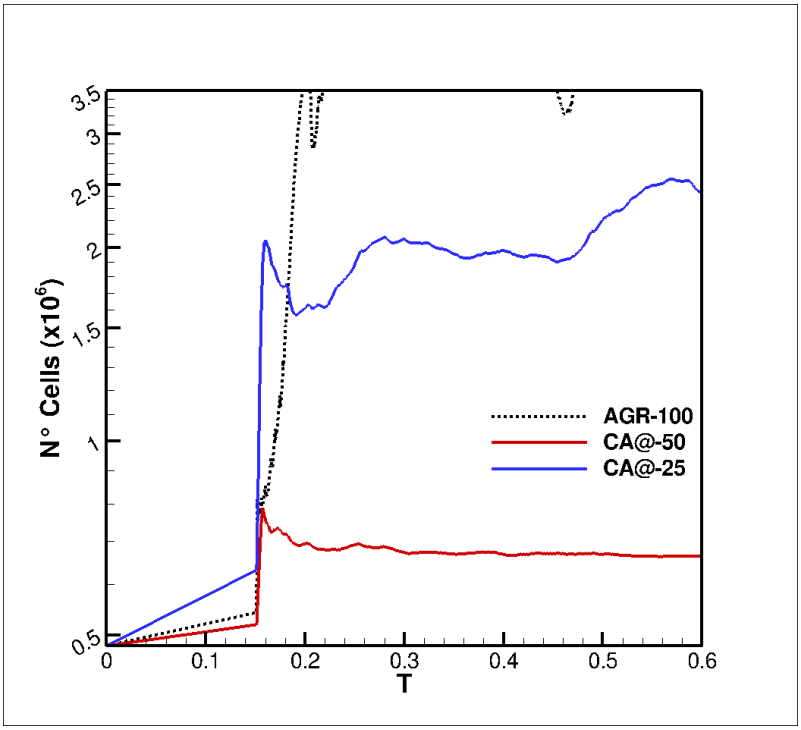
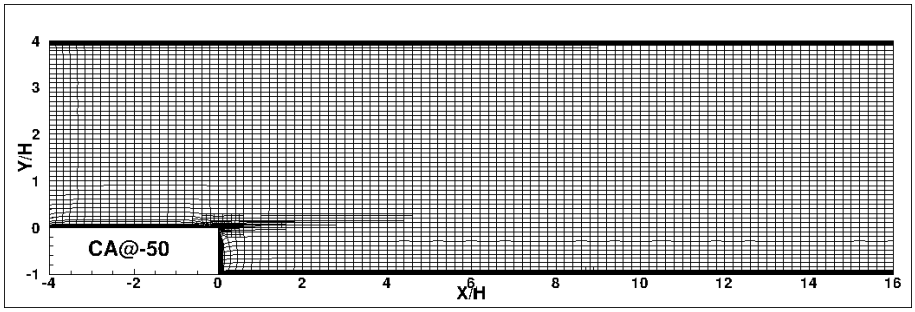
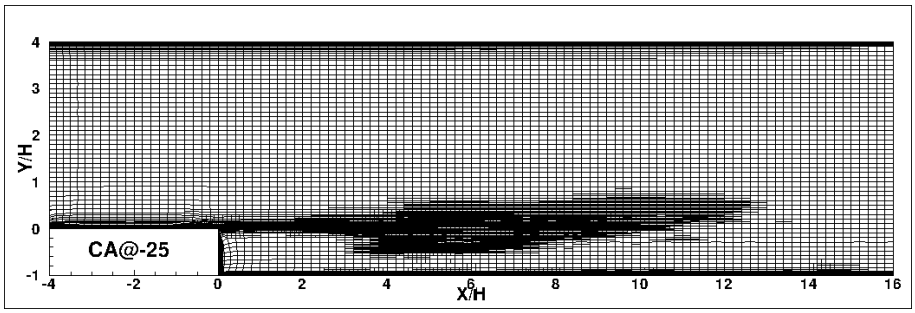


Figure 6.12 – The changes of the number of cells for the two computations *CA@-50* and *CA@-25*



(a) *CA@-50*



(b) *CA@-25*

Figure 6.13 – The final refined meshes for the two computations *CA@-50* and *CA@-25*

thresholds are presented in Figure 6.13. It shows that $T_r = 0.005$ is not small enough for the adaptation to refine the mesh even in the free shear layer region. With $T_r = 0.0025$ however, in the final mesh, refined grids are seen around the shear layer, in parts of the reattachment and the relaxation regions and also close to the walls. The evolution of these refined grids is much less pronounced during the computations (Figure 6.14) compared to the extreme mesh changes found at different instants for the computation *AGR-100* (Figure 6.8). In addition, as the computation continues the recirculation zone gets larger because the effects of the transient solutions on the averaged solution decrease, so the local averaged solution is changed less. Thus, the mesh in the recirculation zone becomes coarser.

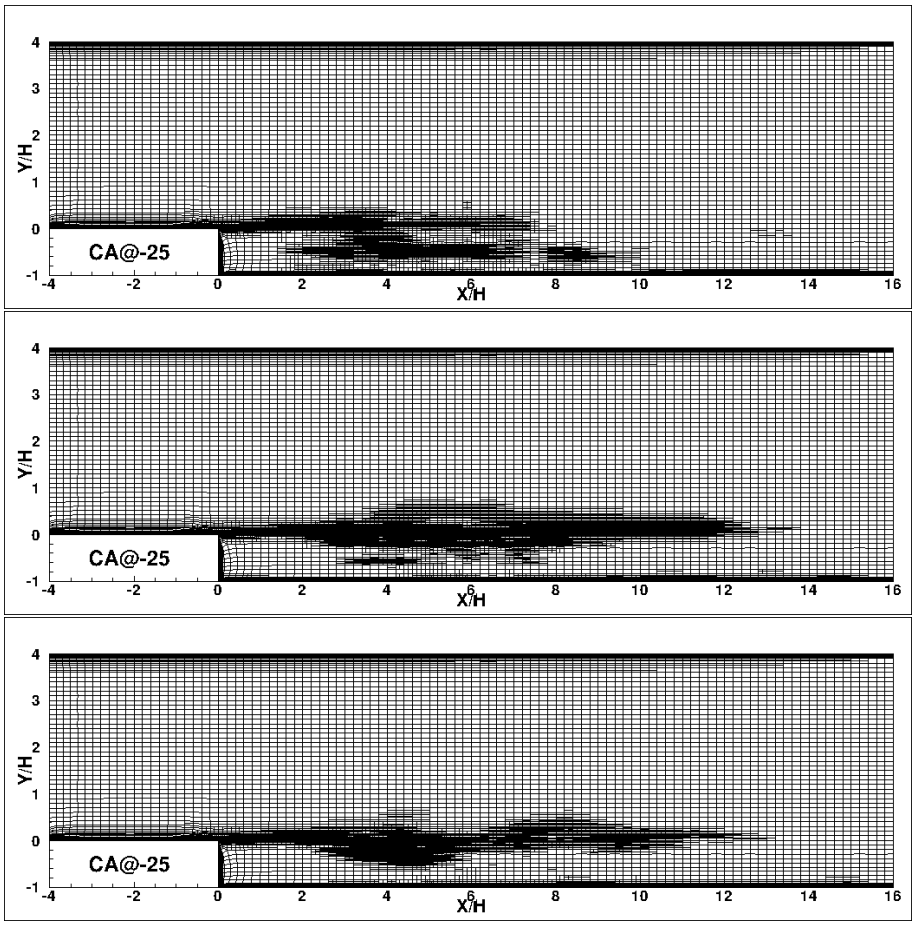
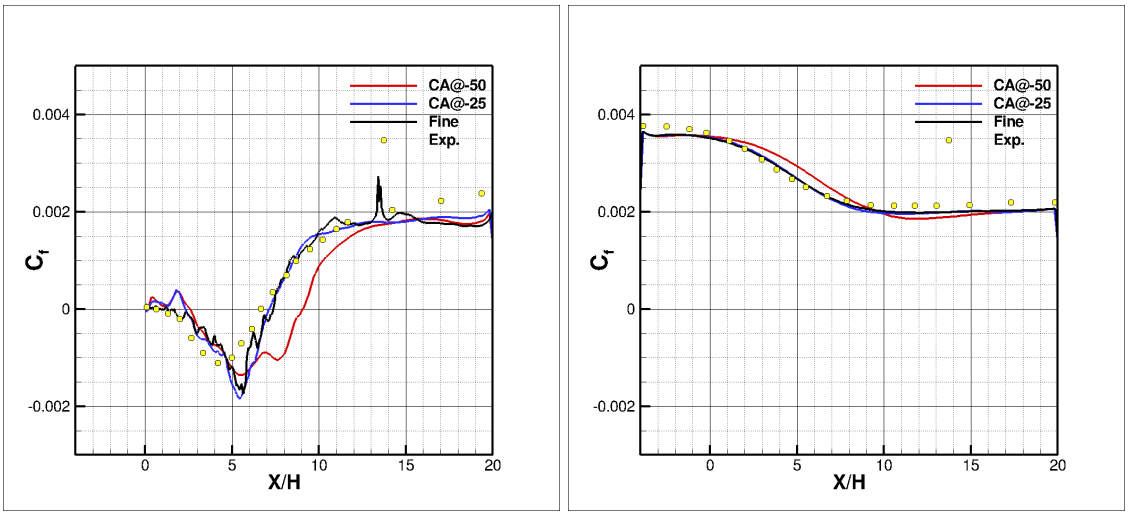


Figure 6.14 – Mesh evolution of *CA@-25* during the refinement process for the instants $T = 0.33, 0.42$ and 0.51

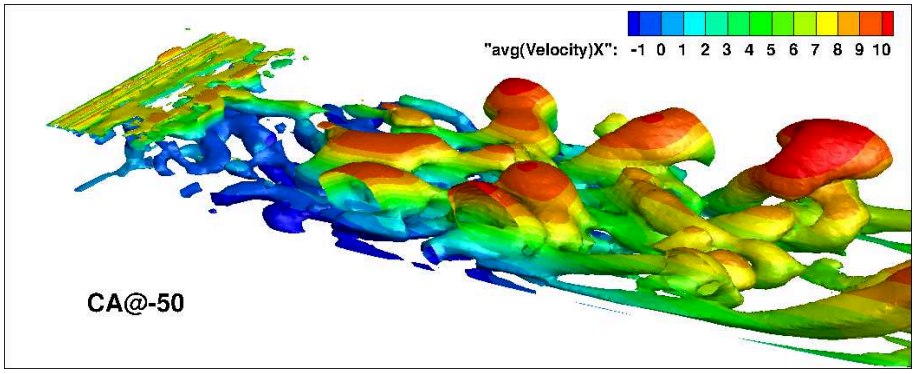
The mean friction coefficient distribution along the walls for these two simulations is presented in Figure 6.15. It shows that the solution is improved by reducing the threshold to $T_r = 0.0025$, especially in the reattachment and the relaxation regions where the refinement is mostly performed. The reattachment length for the computation *CA@-25* is predicted around $X/H = 7.14$ and the difference with experimental data, from 25% of the computation *AGR-100*, goes down to a bit more than 7%. Before the reattachment, the results for the lower wall match the solutions of the previous computations and are overestimated compared to the experimental data.

The iso-surface Q criterion at the end of the simulation which is presented in Figure 6.16 also confirms that $T_r = 0.005$ is not low enough, so the adapted mesh cannot capture the flow features. By reducing the threshold and increasing the refinement, smaller turbulent

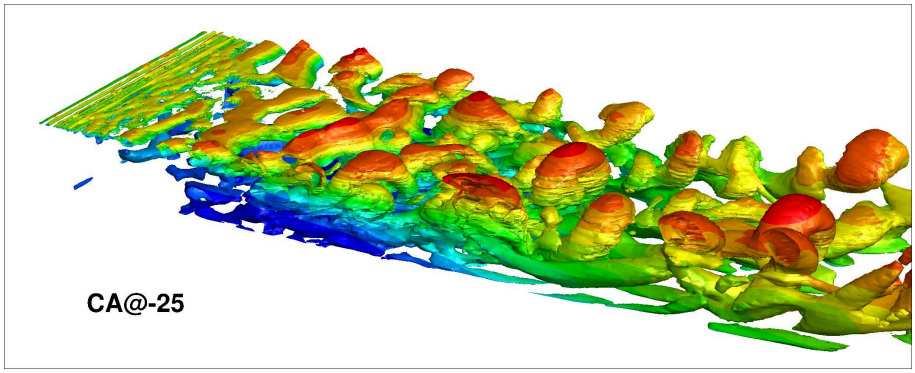


(a) Lower wall (b) Top wall

Figure 6.15 – Mean friction coefficient distributions along the (a): lower wall; (b): top wall for the two computations *CA@-50* and *CA@-25*



(a) *CA@-50*



(b) *CA@-25*

Figure 6.16 – Iso-surface of the second invariant $Q = 2500$ for simulations *CA@-50* and *CA@-25*, colored by averaged streamwise velocity

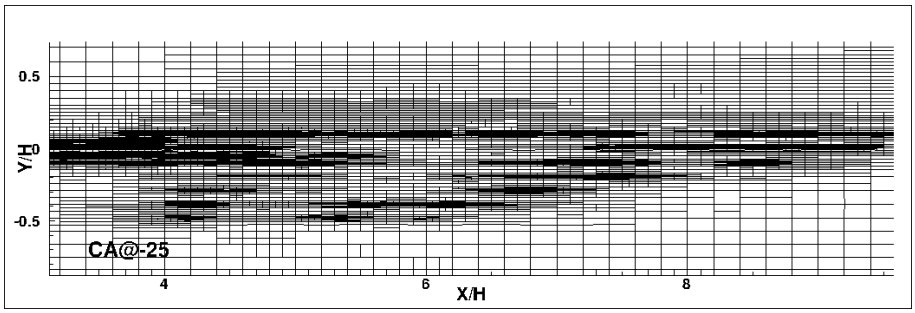


Figure 6.17 – The irregularities of the final refined mesh for the computation CA@-25

structures especially in the reattachment region are captured. However, the small eddies in the recirculation zone are not captured due to the coarse grids in this region (compare with Figure 6.9).

Despite the somewhat acceptable results of adaptive refinement based on averaged solutions compared with the reference simulations, the use of this averaging approach is not possible without modifications in the interpolation process. Zooming into the refinement regions shows that the mesh is not regular (Figure 6.17). These irregularities come from the interpolation process for transferring the averaged solution from the coarse grid to the fine grid, during the mesh refinement. Since the interpolation is linear (C_1), the second derivative of the interpolated solution is discontinuous on the boundaries of the coarse cells and thus those regions are refined excessively. Although AGR based on averaged solutions can predict the solution with fewer mesh changes and cheaper computation (but with some degree of error), the irregularities in the mesh make it invalid and this averaging approach cannot be used unless a higher-order interpolation is performed.

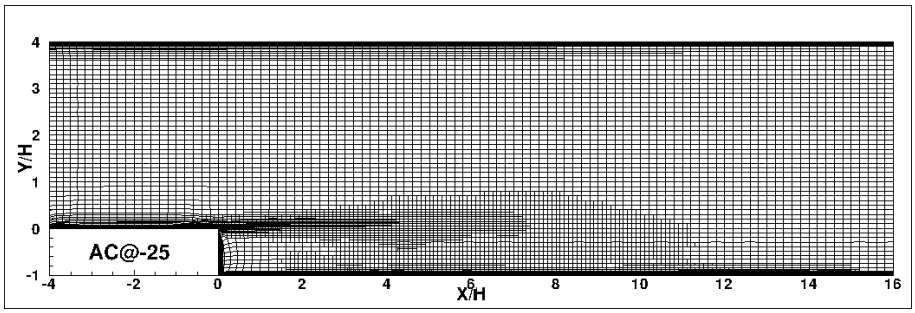
6.3.3 Average of the criterion (AC)

The second approach, in which the refinement criterion is defined from the instantaneous solutions at each time step and the averaging is applied over these computed criteria, is considered here. Thus, the beginning of the simulation code is changed to «*AC*», for Average of the instantaneous Criterion. The simulation conditions remain the same as the previous simulations. Since the threshold of $T_r = 0.005$ was found to be not low enough for AGR to refine the mesh, the first simulation for this approach uses $T_r = 0.0025$.

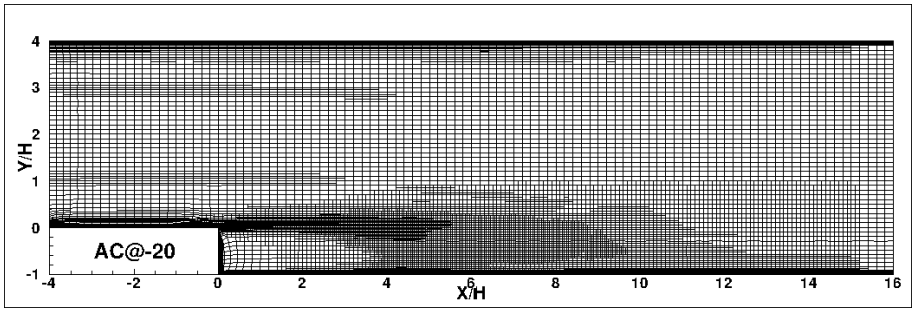
The first effect of averaging over the criterion is the removal of the mesh irregularities (Figure 6.18a). The mesh density is distributed smoothly from the coarse-grid parts to the regions where the higher mesh resolution is required. As time passes and turbulence spreads downstream, the refined regions become larger (Figure 6.19). Like the first averaging approach, the refinement is performed around the shear layer, in the reattachment region, in part of the relaxation region and in the vicinity of the walls. The mesh in the recirculation zone remains coarse due to the low flow rate.

The refinement based on the average of the criterion leads to a regular refined mesh, but with the same threshold and simulation conditions as the case in which the averaging is applied over the flow fields, the final mesh is coarser (see Figure 6.20 and Table 6.3). In addition, the variation in the number of cells during the computation is not the same as before. The figure shows that the increase in the cell number after starting the refinement is less than before. The rate of increase is gradually reduced until the mesh converges to a constant number of cells which is almost half the previous case (1.3 million cells).

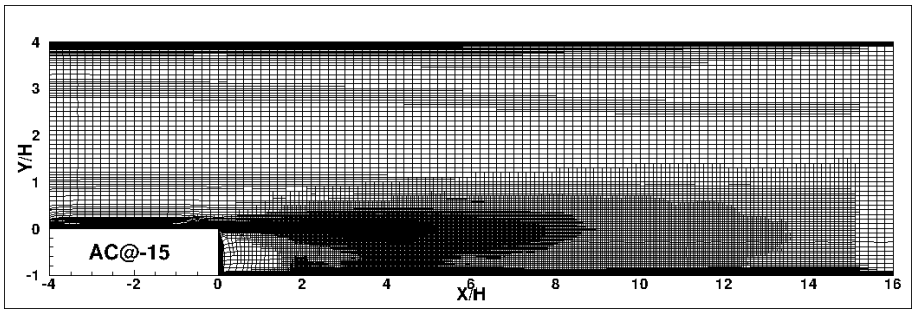
The distance between the mean friction coefficient of the computation *AC@-25* and the experimental data increases in the reattachment and the relaxation regions (Figure 6.21)



(a) $T_r = 0.0025$



(b) $T_r = 0.0020$



(c) $T_r = 0.0015$

Figure 6.18 – The final refined meshes for the computations *AC@-25*, *AC@-20* and *AC@-15*

compared to the other averaging approach. The error in the prediction of the reattachment length (Table 6.3) also increases to more than 18%. So while the refined mesh of the simulation with AGR based on an averaged criterion is regular, the errors in the solutions are higher than before. This can be the result of the coarser final mesh compared to the other case.

To increase the resolution of the final mesh, lower thresholds can be used. Thresholds $T_r = 0.002$ and 0.0015 are tested (Table 6.2). As a result, the mesh becomes finer particularly in the shear layer and the reattachment regions (Figures 6.18b and 6.18c). In the relaxation region, the mesh density increases until the end of the refinement limiting box (Section 3.3.3) at $X/H = 15$. Finally, the grids are finer in the vicinity of the walls and for $T_r = 0.0015$, the edges of the boundary layers are fully captured by AGR. Due to this increase in the refinement, the mesh contains almost 2.5 million cells for simulation *AC@-20* and 10 million cells for *AC@-15* (Figure 6.20) at the end of the computation, and the evolution of the cell numbers indicate that these simulations may not be fully converged.

On one hand, a high-resolution mesh is undesirable in terms of computational cost. On the other hand, such a mesh increases the accuracy of the solutions compared to simulations with higher thresholds. The error in the reattachment length for *AC@-20* is a bit more

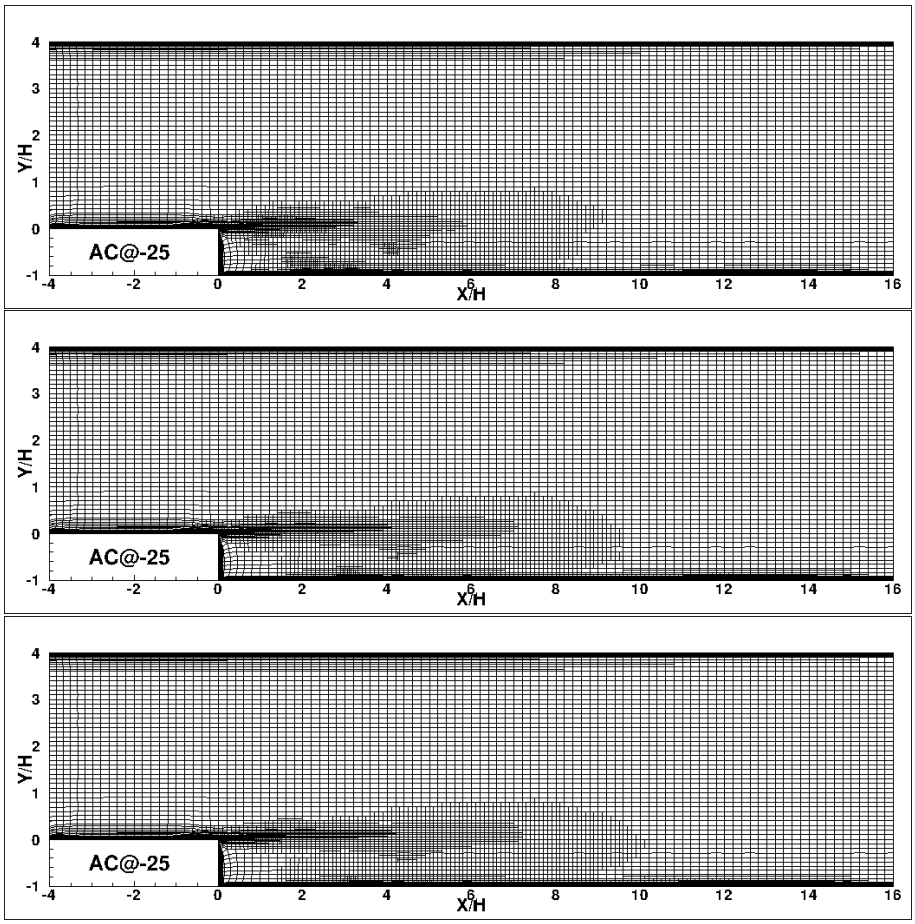


Figure 6.19 – Mesh evolution of *AC@-25* during the refinement process for the instants $T = 0.33, 0.42$ and 0.51

than 5% with only 2.5 million cells. However, the significant increase in the cells number due to the reducing the threshold to 0.0015 in simulation *AC@-15* leads to 3.6% error of the reattachment length (Table 6.3).

The distribution of the mean friction coefficient is also affected by the increase in the mesh resolutions. As the threshold is lowered, the solutions get closed to the experimental data especially in the reattachment and the relaxation regions (Figure 6.21). Figure 6.22 shows how the reduction of the threshold leads to more creation of turbulence in the entire computational domain. For simulation *AC@-15*, the vortices in the shear layer are better represented (Figure 6.22c). The rolling up of the vortex sheets can be detected. The small-size structures of the parts of the recirculation zone which are close to the reattachment point, are also captured. However, even by lowering the threshold, AGR was not able to capture small eddies in the forward recirculation zone and the local mesh remains coarse. This may be also the reason for the overestimated solution of skin friction in this zone.

In conclusion, with respect to *AGR-100*, the average-based refinement leads to a limited variation of mesh size in space and time as long as the averaging applied over the criterion. To obtain a compatible solution with the experimental data, a significant number of cells is still needed. Reduction of the threshold results in more refined grids in the recirculation region, but the smallest eddies are still not captured. Although the results of *AC@-15* with around 10 million cells are in good agreement with the experimental data, the solutions are still affected by the lack of convergence of the mesh. This is a consequence of using the

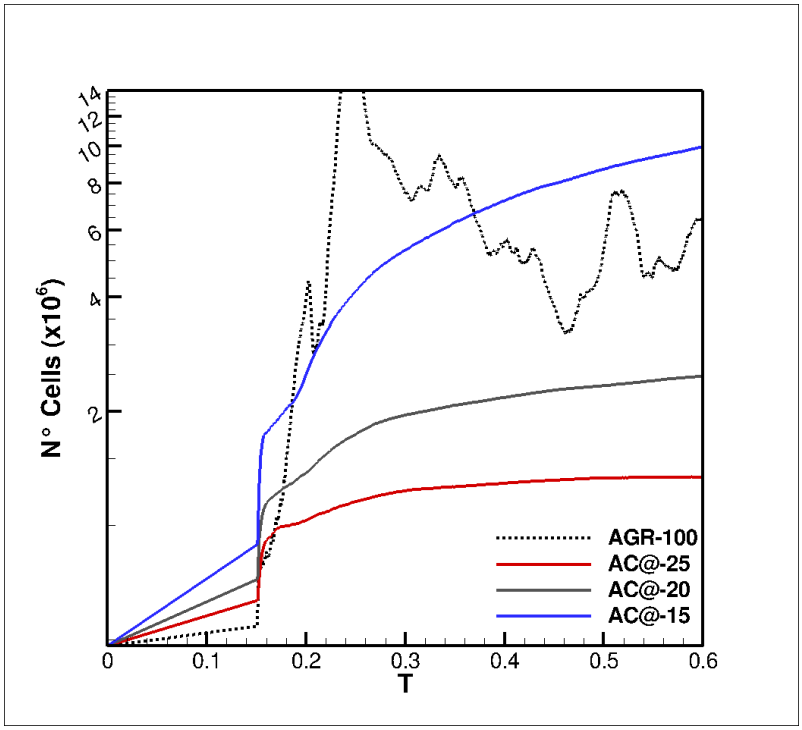
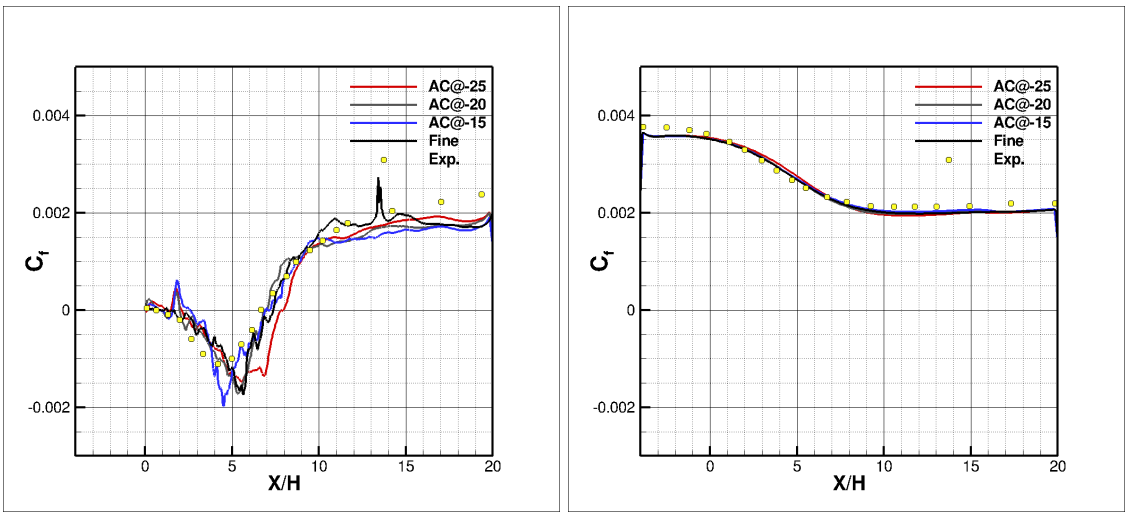
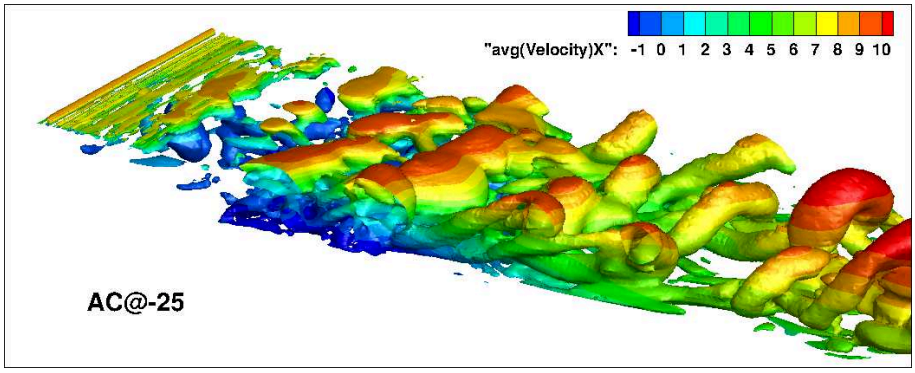


Figure 6.20 – The changes of the number of cells for the simulations with the average of the criterion

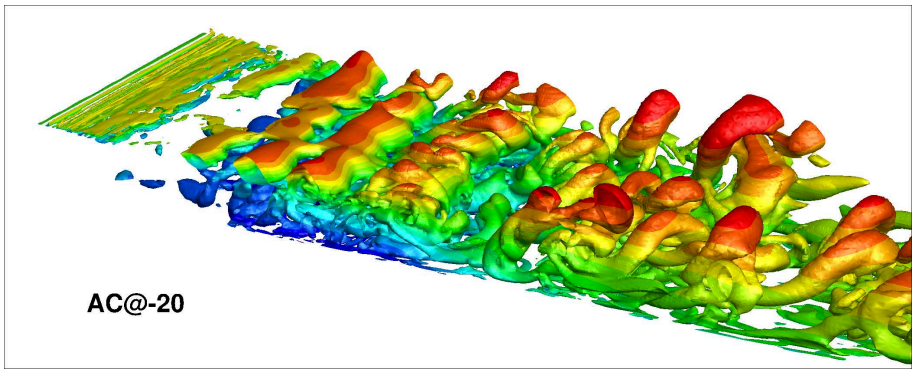


(a) Lower wall (b) Top wall

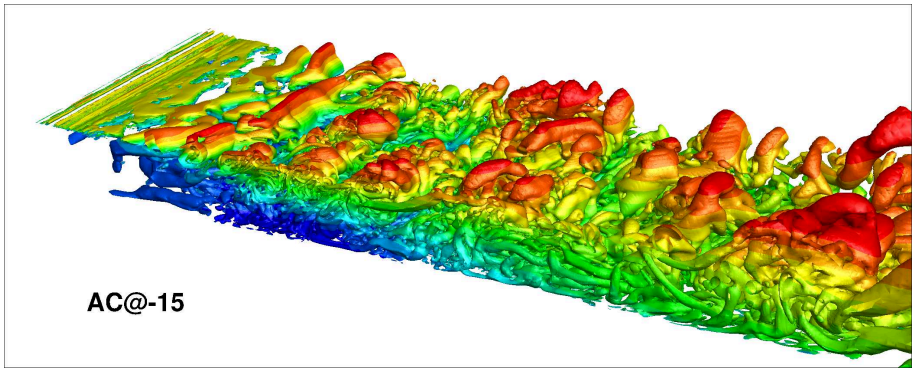
Figure 6.21 – Mean friction coefficient distributions along the (a): lower wall; (b): top wall for the two computations *CA@-25* and *AC@-25*



(a) *AC@-25*



(b) *AC@-20*



(c) *AC@-15*

Figure 6.22 – Iso-surface of the second invariant $Q = 2500$ for simulations *AC@-25*, *AC@-20* and *AC@-15*, colored by averaged streamwise velocity

entire computational time as an averaging interval, which implies that the speed of change in the mesh towards convergence gets slower as the simulation runs for a longer time.

6.4 Averaging interval analysis

In the previous computations, the final refined meshes were based on the averaging over the entire computational time. The final averaging interval is 7500 time steps which is close to one complete period of the flapping motion (7000 time steps, see Section 5.4). This diminishes the effect of low-frequency flow features on the averaged solution and reduces the rate of change of the averaged solution, which causes the mesh adaptation to occur slowly, as seen above. Reducing the size of the averaging interval increases the rate of change in the averaged solution (Chapter 5), since a smaller interval increases the effects of the high-frequency unsteady flow features on the averaged solutions. In addition, any change in the flow conditions, such as an increase or a decrease in the inlet velocity, changes the averaged solution sooner. This faster evolution of the averaged solution allows the AGR to be more reactive and adapt the mesh quicker.

In this section, for the previous average-based simulations, shorter sliding averaging intervals are tested. Two simulations *AC50-25* and *AC01-25* are performed in this section. In both simulations, the refinement is applied based on the average of the criterion, and the threshold is $T_r = 0.0025$. The difference is only in the size of the sliding interval. For the simulation *AC50-25*, the averaging is applied over an interval which contains 50 refinements. There are 25 time steps between two consecutive refinements, therefore the size of the interval is 1250 time steps, which corresponds to 2.5 periods of the shedding mode. In Chapter 5, it was seen that averaging over this interval is sufficient to diminish the effect of the high-frequency flow features in the shear layer. For the other computation *AC01-25*, the averaging interval is one refinement interval or 25 time steps (Figure 6.11c). This short interval is less than one period of K-H instability in the shear layer which is about 100 time steps.

The variation of the number of cells in Figure 6.23 shows an increase in the rate of mesh change. For the *AC01-25*, the changes in the mesh are close to the simulation *AGR-100*, where no averaging is applied. Although in *AC01-25* the refinement is not applied as frequently as the simulation *AGR-100* - every 25 time steps compared with every 2 time steps - and is based on the average of 25 solutions instead of the instantaneous solutions, the threshold is much lower so a larger amount of refinement is allowed. For this reason the computation was expensive and was terminated before it finished.

In simulation *AC50-25*, the refined mesh at the end of the computation contains twice the number of cells of the equivalent simulation with averaging over the entire computational time (Table 6.3). Applying the refinement based on a sliding interval of 1250 time steps increases the effects of the flow features like the shedding mode on the averaged solution. Therefore, the AGR refines the mesh more to follow these unsteady features. Most of this refinement is applied in the reattachment and the relaxation regions (Figure 6.24) where these unsteady features are more pronounced; this makes the mesh in these regions less smooth.

The shorter interval also has an effect on the speed of the mesh convergence. Comparing the final mesh of the two simulations *AC@-25* and *AC50-25* in Figure 6.24 shows the same refinement in the shear layer region. However, the refinement in parts of the recirculation zone for *AC@-25* no longer exists in the final refined mesh of *AC50-25* and, on the contrary, the relaxation region is refined more. This implies that a shorter interval increases the convergence speed of the adaptation compared to the refinement based on the entire computational time. The evolution of the mesh in several instants of the computation for

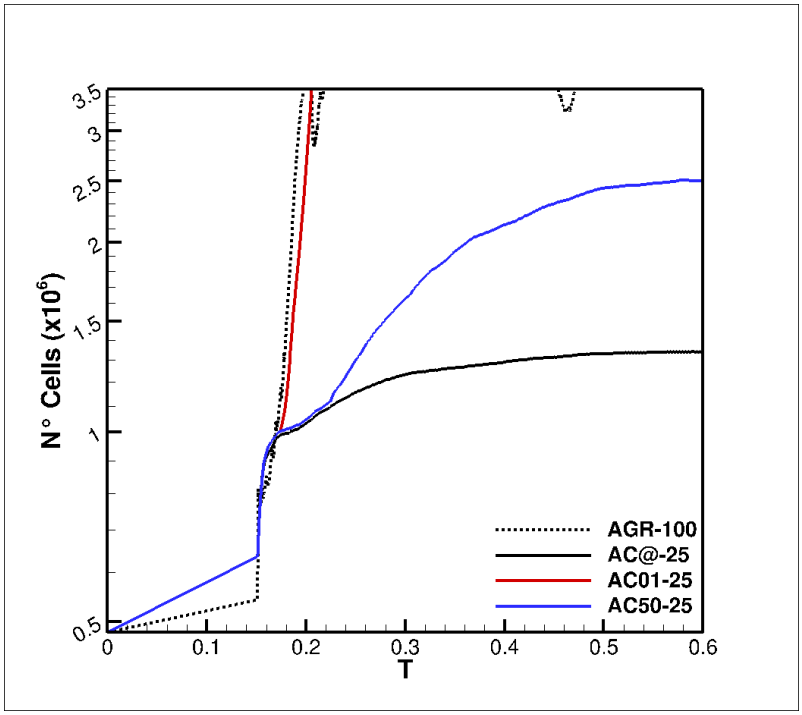
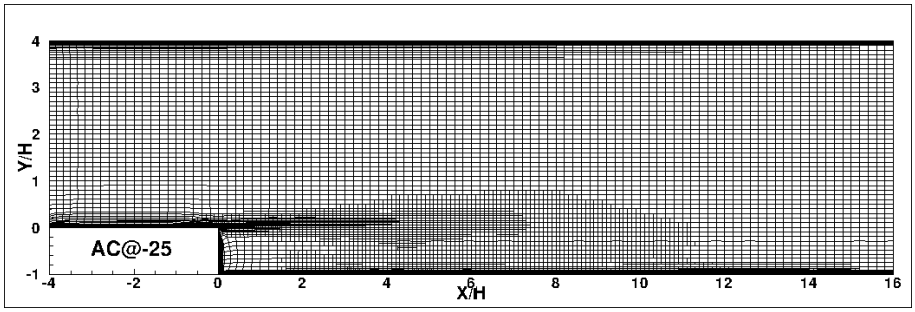
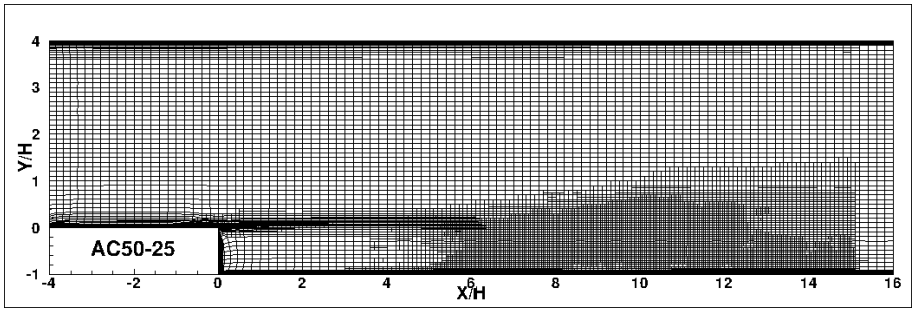


Figure 6.23 – The changes of the number of cells for the two computations *AC01-25* and *AC50-25*, as well as *AC@-25*



(a) *AC@-25*



(b) *AC50-25*

Figure 6.24 – The final refined meshes for the two computations *AC50-25* and *AC@-25*

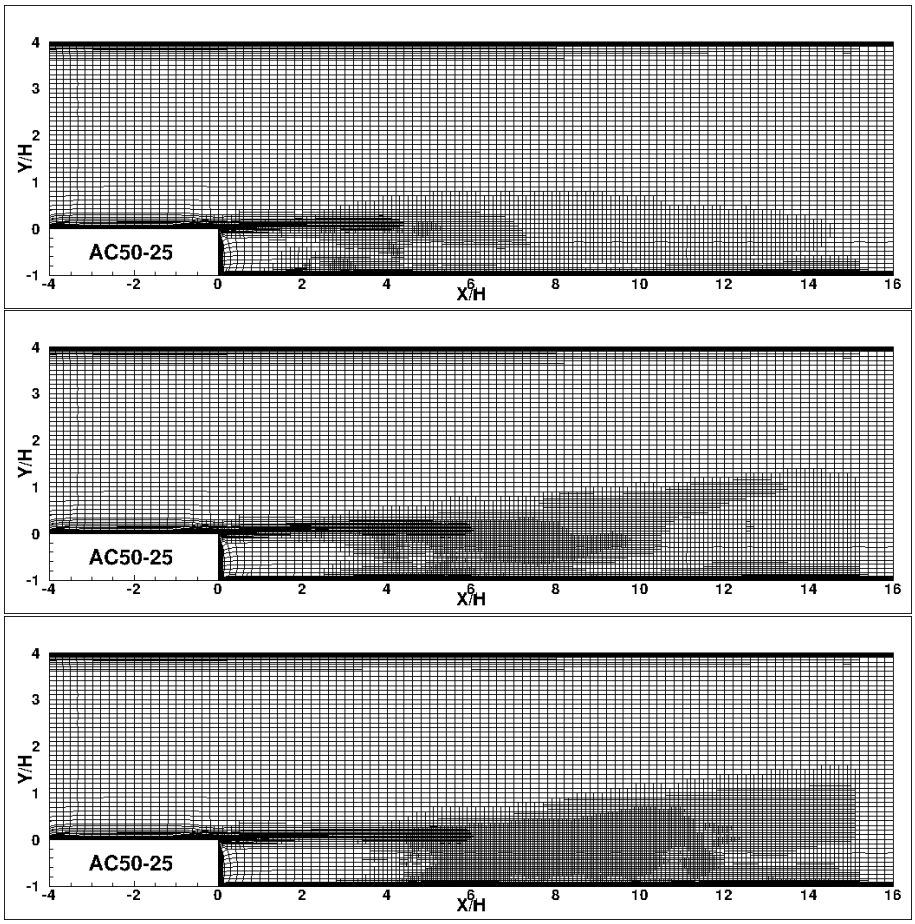
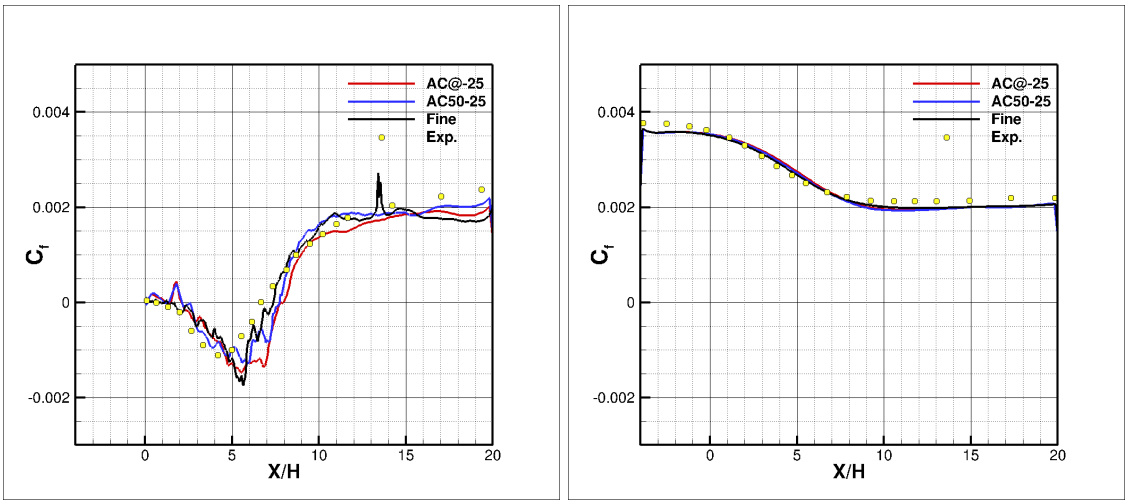


Figure 6.25 – Mesh evolution of *AC50-25* during the refinement process for the instants $T = 0.33, 0.42$ and 0.51

the simulation *AC50-25* (Figure 6.25) shows how fast the mesh undergoes changes toward convergence compared to the simulation *AC@-25* (Figure 6.19), particularly for the regions like the shear layer and the recirculation zone which are under the direct influence of the shedding mode due to the short averaging interval length.

The distribution of the mean friction coefficient in Figure 6.26 shows improvement in the reattachment and relaxation regions for the lower wall. In addition, the error of the prediction of the reattachment length (Table 6.3) is slightly smaller than *AC@-25* even though the refinement in the recirculation region is less. The iso-surface Q criterion of this simulation (Figure 6.27) is very similar to *AC@-25* (Figure 6.22a), except for the relaxation region in which more smaller-size turbulent structures are created due to the local refinement in this region.

Generally, there is a limitation on the sliding window length, due to the required memory to save the solutions. Although using sets of averaged values instead of saving all the solutions reduced significantly the required memory, it is still expensive compared to the standard ISIS-CFD simulation. A shorter interval causes the mesh adaptation to react faster to the flow evolution, so the convergence of the mesh is obtained faster, contrary to the simulations with the averaging interval of the entire computational time in which the adaptation of the mesh is performed slowly. A second effect is that, due to the increase of the effects of flow features like the flapping motion and the shedding mode on the averaged solution, more refinement and more mesh changes appear in the reattachment and the relaxation regions. This may or may not be a desired effect.



(a) Lower wall (b) Top wall

Figure 6.26 – Mean friction coefficient distributions along the (a): lower wall; (b): top wall for the computation *AC50-25*

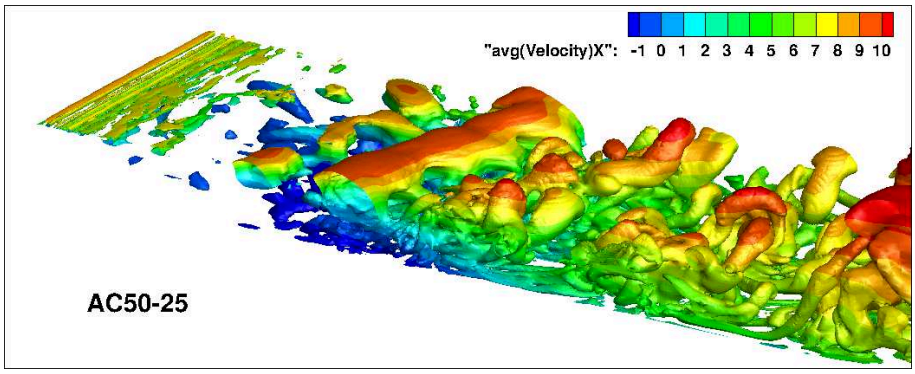


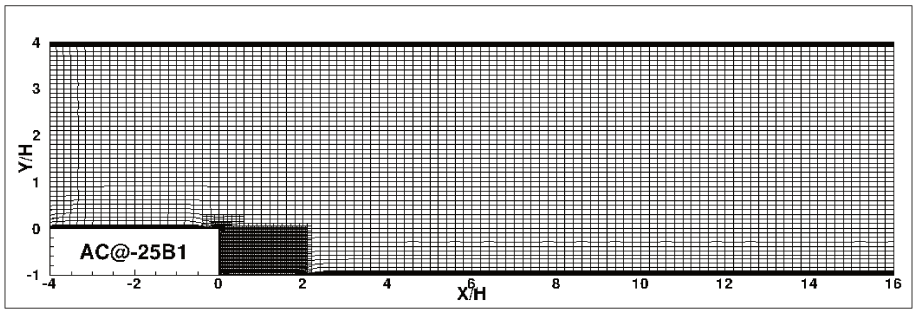
Figure 6.27 – Iso-surface of the second invariant $Q = 2500$ for *AC50-25*; colored by averaged streamwise velocity

6.5 Effect of the recirculation zone

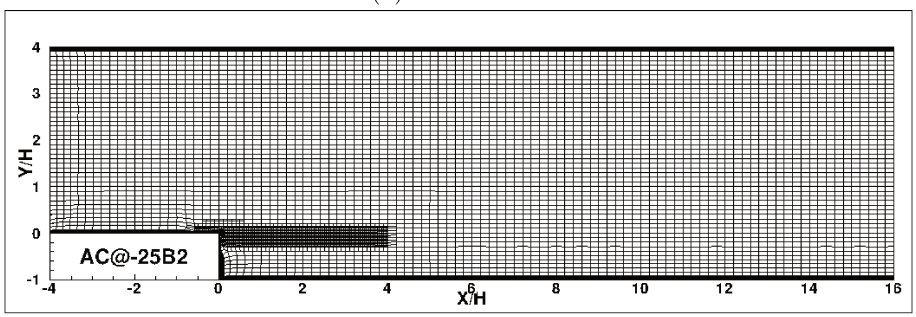
In all the previous simulations with AGR, no matter which averaging approach is performed or what the size of the averaging interval is, the local grids in the recirculation zone are coarse. The reason for this is the inability of the refinement criterion to capture small turbulent eddies in the recirculation zone. Since the rate of flow in this zone compared to other parts of the domain is very low, the local value of the refinement criterion which is the second derivative of the flow field, is not large enough to lead to any refinement in the recirculation zone. This is exactly what was observed by Castro-Díaz et al [16]. They proposed a scaled refinement criterion to increase the refinement in the recirculation zone, especially for the secondary vortices in the corner of the step. However, the principle of AGR with Hessian-based criteria implies that low-gradient flow has no influence on the rest of the domain, so it can be resolved with coarse meshes.

The question here is, what the influence of a coarse mesh in the recirculation zone is, for the BFS flow. Two possible answers can be assumed for this question: first, there may be an effect of the small eddies in the recirculation zone on the shear layer. These small eddies recirculate into the shear layer and may cause the shear layer to be destabilized sooner, which has an effect on the location of the reattachment point on a coarse mesh, if

these eddies are missing. The second answer could be the slow generation of LES content in time. Until the mesh is refined around the shear layer and becomes fine enough to switch the turbulence model from RANS to LES, the separation is modelled by RANS and there is no resolved solution available in the region. The virtual fluctuations (see Section 4.2.1) at the inlet boundary cannot be used, because the intial mesh is too coarse and any upstream resolved solution is dissipated before reaching the step. Moreover, the fluctuations would not be intense enough to be followed by AGR and transferred to the separation. So when the LES behavior of the turbulence model is not activated at the step, there is no LES content anywhere in the domain. Thus, with an initial coarse mesh, the solution of the downstream regions develops slowly.



(a) *AC@-25B1*



(b) *AC@-25B2*

Figure 6.28 – Initail coarse mesh with different refinement boxes for *AC@-25B1* and *AC@-25B2*

To understand the importance of each assumption and to see which one has a greater impact on the downstream solutions, two simulations called *AC@-25B1* and *AC@-25B2* are performed which are based on *AC@-25* in Section 6.3. The only difference is in the initial coarse mesh which is used. Two different refinement boxes are used to study the effects of the small turbulent eddies in the recirculation zone and to see how the faster representation of the shear layer will affect the solution. For *AC@-25B1*, a refinement box is added in a part of the recirculation zone and the shear layer (Figure 6.28a), and for *AC@-25B2*, the refinement box covers only the shear layer (Figure 6.28b). For the latter refinement box, since the recirculation region flow moves in circles, any small-scale turbulence is removed by the coarse grid below the step. Thus, in the second simulation, only the effect of the faster generation of the LES content is investigated. The refinement conditions are kept the same as before; refinement is based on the average of the criterion, the averaging interval is set to the entire computational time and the threshold is 0.0025.

By using the refinement boxes, more turbulent structures are created (Figure 6.30). The evolution of these structures is detected by the averaged refinement criterion and as a result a finer mesh compared to the computation without refinement box is obtained; the final converged meshes for both simulations have around 2 million cells (Figure 6.29).

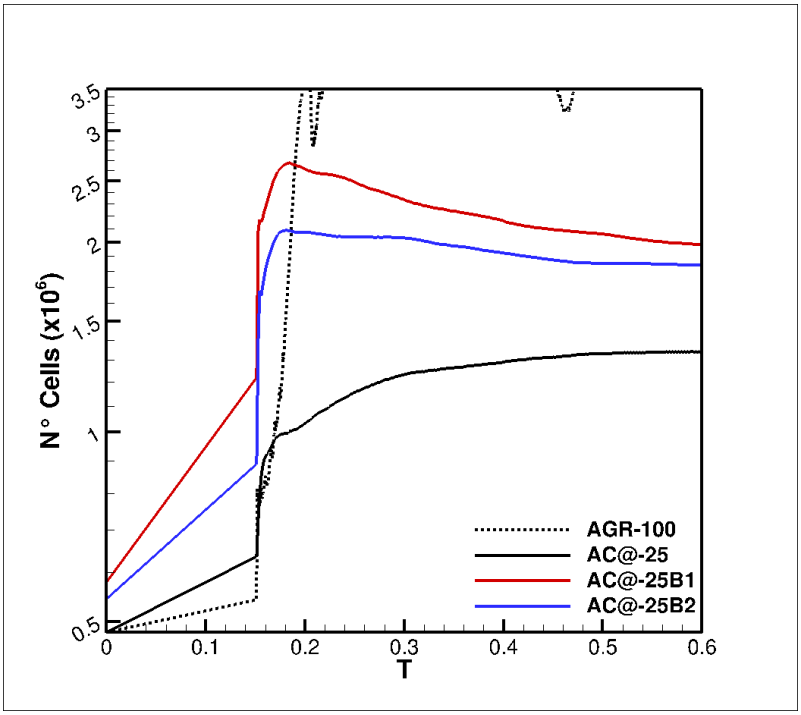


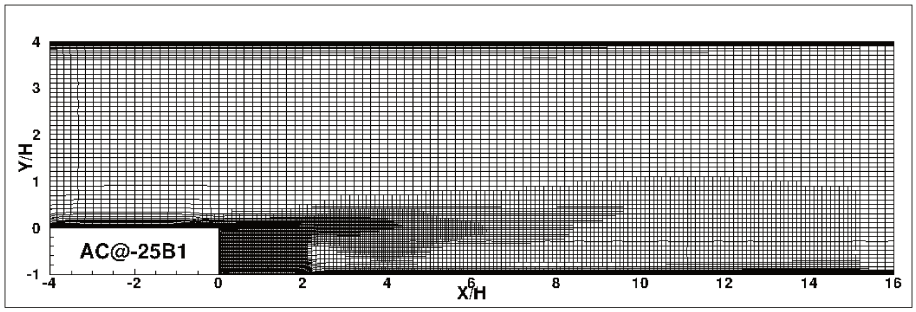
Figure 6.29 – The changes of the number of cells for the computations *AC@-25B1* and *AC@-25B2*, as well as *AC@-25*

This quick increase in the number of cells in both the simulations is indicative of the faster transient due to the initial mesh refinement in the shear layer.

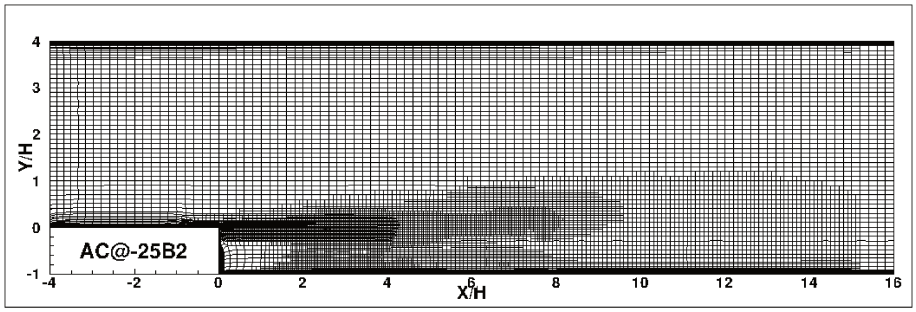
The similarity of the two final meshes in Figure 6.30 also confirms that it is the removal of the transient effect which leads to the high-resolution refined meshes in the relaxation region, compared to the final mesh of *AC@-25* (Figure 6.18a). The computations start with a fine grid behind the separation point and around the shear layer. The resolution of the mesh in these regions is high enough to activate the LES behavior of the turbulence model even during the first 2500 time steps which are dedicated to removing the transient flow. This is not the case for *AC@-25* which starts with an initial coarse mesh without any refinement box. Thus, the presence of the refinement box is the shear layer region gives the turbulence model enough time to generate resolved content to represent the shear layer. The propagation of this turbulent content toward the downstream region causes more refinement in the reattachment and the relaxation regions.

The distribution of the mean friction coefficient in the reattachment and relaxation zones is improved for both simulations as a result of the better representation of the shear layer by adding the refinement boxes (Figure 6.31). These results are in good agreement with the experimental results. However, the reattachment length is underpredicted for the simulations *AC@-25B1* and *AC@-25B2* and is reduced to $X/H = 6.5$ and $X/H = 6.3$ respectively (Table 6.3). The mean friction coefficient is similar for the two simulations and in Figure 6.32, the flow structures for both simulations are almost the same, except in the recirculation zone of *AC@-25B1* where more small-scale structures are captured. Considering discussed results for both simulations, the improvements seem to be mainly the result of the refinement around the shear layer which leads to an increase in the convergence rate in the recirculation zone. In contrast, the small-scale eddies in the recirculation zone have minor effects on the improvement of the solutions. The underestimation of the recirculation length may be due to the long averaging interval.

In general, although the small eddies in the recirculation zone play some role in the

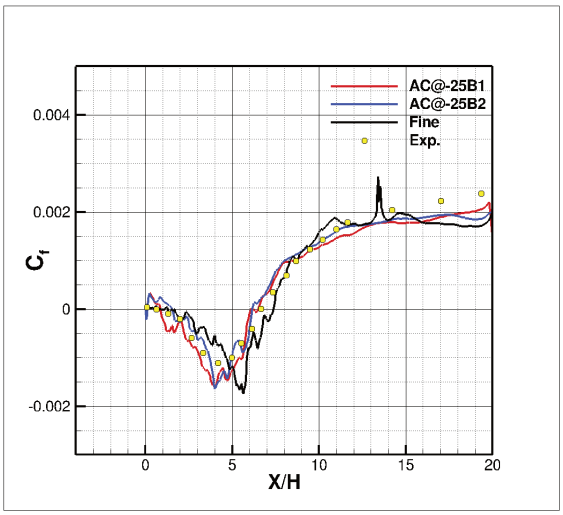


(a) *AC@-25B1*

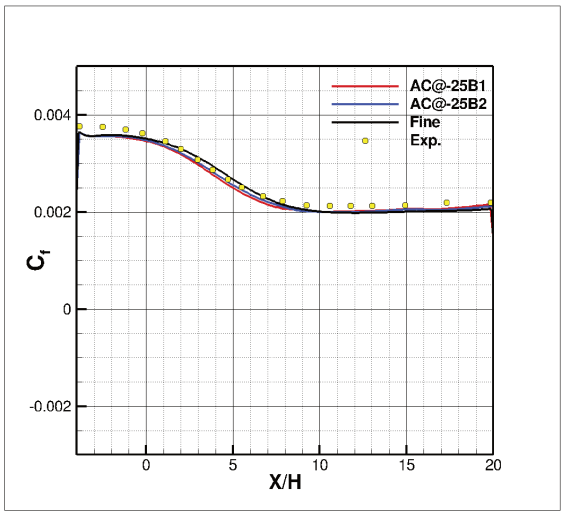


(b) *AC@-25B2*

Figure 6.30 – The final refined meshes for the computation *AC@-25B1* and *AC@-25B2*

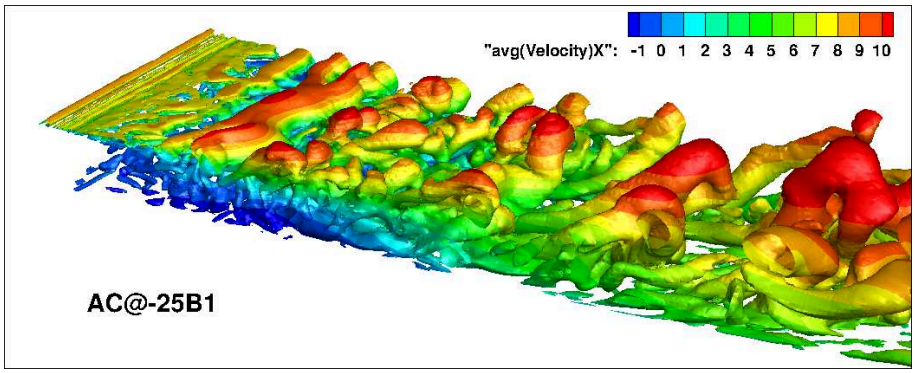


(a) Lower wall

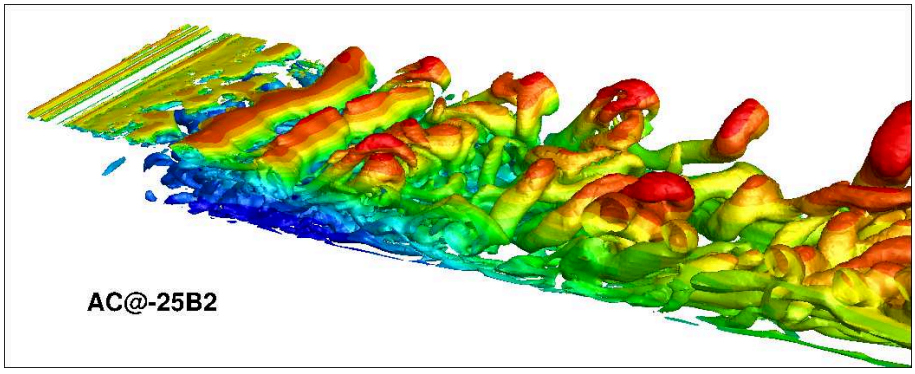


(b) Top wall

Figure 6.31 – Mean friction coefficient distributions along the (a): lower wall; (b): top wall for the computations with refinement boxes



(a) *AC@-25B1*



(b) *AC@-25B2*

Figure 6.32 – Iso-surface of the second invariant $Q = 2500$ for simulations with refinement box, colored by averaged streamwise velocity

improvement of the solution (particularly the mean friction coefficient), for the entire flow field, rapid resolution of the shear layer which is made possible by a refinement box around the separation point and the shear layer, is the most important effective parameter in improving the results. This validates the concept of refining where the most intense flow features occur.

6.6 Conclusion

This chapter studies the refinement of the mesh based on averaged quantities, which limits the mesh changes and therefore reduces the undesirable effects of the coupling between AGR and hybrid RANS/LES. Simply applying the AGR based on the instantaneous solutions every two time steps results in a fast-evolving refined mesh which follows the small structures of the flow. The number of cells increases significantly after the AGR is applied and undergoes a large variation during the computation. However, the local grids in the recirculation zone remain coarse and no resolved solution is created in this region. In addition, large perturbations are imposed into the averaged solutions of the reattachment and the relaxation regions, such as a large error in the prediction of the reattachment length and the mean friction coefficient (Figure 6.10) compared to the experimental data.

Basing the refinement criterion on the averaged solution leads to an improvement in the distribution of the mean friction coefficient and a decrease of the reattachment length error to 7.2% compared to the experimental data (Table 6.3). The changes in the mesh during the computation are limited to a great extent and the computational cost decreases due to the lower resolution of the mesh. However, the final refined mesh contains irregularities

which are the result of a fundamental problem in the process of interpolating the coarse-grid averaged solutions to the fine grids. These irregularities make the mesh invalid and so this averaging approach is impracticable.

The second approach is to compute the refinement criterion from the instantaneous flow fields before each time step and to apply the refinement based on the average value of the criterion. This approach removes the irregularities in the mesh, reduces the variation in the number of cells, and at the end of the computation leads to a lower-resolution mesh compared to the first averaging approach. This solution is still far from the simulation with the fine mesh and the experimental data, but it is improved by lowering the threshold. However the improvements in the solution are accompanied by an increase in the mesh size and therefore the computational efforts. To have an optimal threshold for these simulations, a compromise between the accuracy of the solution and the computational efforts should be reached.

So far in all simulations, the time-averaging is applied over the entire computational time (7500 time steps). When the interval size is reduced to 1250 time steps which is equal to 2.5 periods of the shedding mode, the influence of the flapping motion and the shedding mode results in an increase in the local resolution of the mesh in the reattachment and the relaxation regions (Figure 6.25). In addition, the adaptation of the mesh according to the flow is performed faster by using a shorter interval, thus it can be implied that the mesh converges faster for this case. The prediction of the reattachment length and the distribution of the mean friction coefficient, particularly in the reattachment region are slightly better than the solutions of the simulation *AC@-25* in which the averaging is applied over the entire computational time. In all performed simulations with AGR using an averaging interval of the entire computational time, a long computational time is needed for the recirculation region to grow. Thus, the sliding window approach is useful to prevent underprediction of the reattachment length.

Although the convergence rate in recirculation zone is accelerated by using a sliding window, a locally coarse mesh in the recirculation zone is a common point among all the simulations regardless of the applied averaging interval. Two simulations with a refinement box at different locations are performed to study the effects of the small eddies in the recirculation zone as well as the importance of the transient mesh adaptation for the shear layer. Comparing the results for both simulations *AC@-25B1* and *AC@-25B2* shows that the improvements in the solutions, such as the prediction of the reattachment length and the mean friction coefficient, are mostly due to a better representation of the shear layer and the generation of the resolved solutions in the region and the effects of the small eddies in the recirculation zone on improving the results have a less importance.

In summary, using AGR based on the average of the computed criterion has an advantage over the simulation without averaging, in terms of the accuracy of the solutions and also the computational cost. For any given flow, the average-based adaptive grid refinement based on the averaged of the criterion can be used. To ensure that the threshold value is not too low to cause excessive refinement or too high to have too coarse grids, the variation of the number of cells during the first several refinements should be monitored. The averaging interval is set based on the flow behavior. For a flow whose behavior and features are not fully understood, averaging over the entire computational time would be a safer choice. It does not require much memory and also severe changes in the flow will have a more limited impact on the refinement process. However, if the behavior of the flow is understood, a sliding averaging window can be used to obtain faster convergence. The length of the sliding window can be selected based on the frequency of the desired features in the flow. In the case of a low-intensity flow region, like the recirculation zone in the BFS flow, the AGR does not refine the mesh in this region. However, the effects of the

small-scale eddies in this region on downstream are probably minor.

Chapter 7

Realistic test case

The purpose of this chapter is to perform and analyze a realistic test case where hybrid RANS-LES and average-based adaptive refinement function together. The simulation of trailing vortices created by the DTMB 5512 in sideslip is considered as this test case.

The physics of longitudinal vortices have been studied in numerical marine hydrodynamics since the 1980s. The focus today is on the internal structure of these vortices, the core dynamics, and the turbulence. For example, one of the test cases for the 2015 Tokyo Workshop on CFD in Ship Hydrodynamics was the Japan Bulk Carrier, a modern cargo ship with a full hull form and, as a result, two strong separated vortices below the aft hull. Early towing tank experiments for this ship show a high-intensity maximum of turbulence kinetic energy (TKE) in the centers of these vortices, while all RANS simulations for the workshop indicate much less turbulence, with a minimum in the center. Only the turbulence-resolving simulations by the University of Rostock [1] and Centrale Nantes / CNRS [25, 103] were able to predict the turbulence correctly. In later experiments, the high-intensity turbulence kinetic energy in the centers of the vortices is not observed which led to further discussions about the experimental conditions.

However, the same observation of high-intensity TKE was made for the flow around the DTMB 5512 destroyer in sideslip conditions, which is studied in the framework of the NATO AVT-253 project. When subjected to sideslip, this ship creates strong trailing vortices which emanate from the sonar dome at the front and the anti-roll keels on the side of the hull, as well as the aftship. The detailed experiments from IIHR [116] show again a maximum for the turbulence kinetic energy in the cores of these vortices.

An explanation for this phenomenon was proposed by Visonneau et al. [104], based on hybrid RANS/LES simulations of the DTMB 5512 using a fine mesh of 163M cells. They discovered that the main vortex cores, which are traditionally seen as rotating longitudinal cylinders of relatively calm flow, are in fact filled with small-scale coherent vortices created around the regions of flow separation and then absorbed into the vortex core. The influence of these vortices is responsible for the high level of TKE. To represent this highly unsteady flow, a turbulence-resolving model was essential, since the unsteady vortices cannot be represented in the average-flow framework associated with RANS.

However, a particular difficulty of this work was the manual placement of the refinement boxes around the vortex cores to create the fine mesh. This required an iterative construction of the finest mesh, with three or four complete simulations, which is a labor- and computational time-intensive procedure. Thus, the manual creation of fine meshes is a limiting factor for this type of stud, especially when the motion is not aligned with the Cartesian axis.

The objective of this chapter is to investigate the possibility to produce fine-grid simulations for the DTMB 5512 using the developed average-based adaptive refinement.

If this refinement can produce a fine mesh which is adapted to the flow features from an original coarse mesh, the simulation process becomes simpler and more efficient. Average-based adaptive refinement is used in a RANS and a DDES-SST simulation of the DTMB 5512 to investigate the mesh adaptation process for capturing and following the turbulent structures of the flow around this test case. The properties of the test case and the simulation settings are presented in Section 7.1 and 7.2. In Section 7.3, the parameters of the adaptation process such as the refinement criterion, the thresholds and the averaging strategies are discussed for each simulation. The results obtained are used to evaluate the global behavior of the flow structures (Section 7.4). In the same section, the computed parameters such as the force coefficients, the turbulence kinetic energy and the axial velocity of the simulations are compared with the available experimental data from IIHR. In Section 7.5, the local solutions at the center of the vortices are compared both with the experiments and with the simulations on a static fine mesh from [104]. The evolution of the adapted meshes during the computation to reach a converged mesh is considered in Section 7.6. Finally, the chapter ends with a conclusion on the performance of the averaged refinement.

7.1 Test case

The DTMB 5512 is a scaled-down version ($L = 3.048m$) of the DTMB 5415, a model of an early concept for the USS Arleigh Burke class of destroyers. The DTMB 5415 and 5512 are well-known ship models that have been experimentally tested in many different conditions. Experiments in steady sideslip conditions performed at Iowa Institute of Hydraulics Research (IIHR) are reported by Yoon et al. [116]. For sideslip, the hull moves sideways as well as forward. This creates an angle of incidence for the flow which produces a lateral force and can provoke strong trailing vortices.

The case tested here has a sideslip angle of 10° to port and a velocity of 1.53 m/s corresponding to the full-scale cruise speed of 20 knots, which leads to a Froude number $Fr = 0.28$ and a Reynolds number $Re = 4.65 \times 10^6$. The experimental measurements were performed at several axial planes from $X/L = 0.06$ to 1 (Figure 7.1). The available data include forces and moments, as well as tomographic particle image velocimetry (TPIV) measurements of the velocity and turbulence variations in the axial planes.

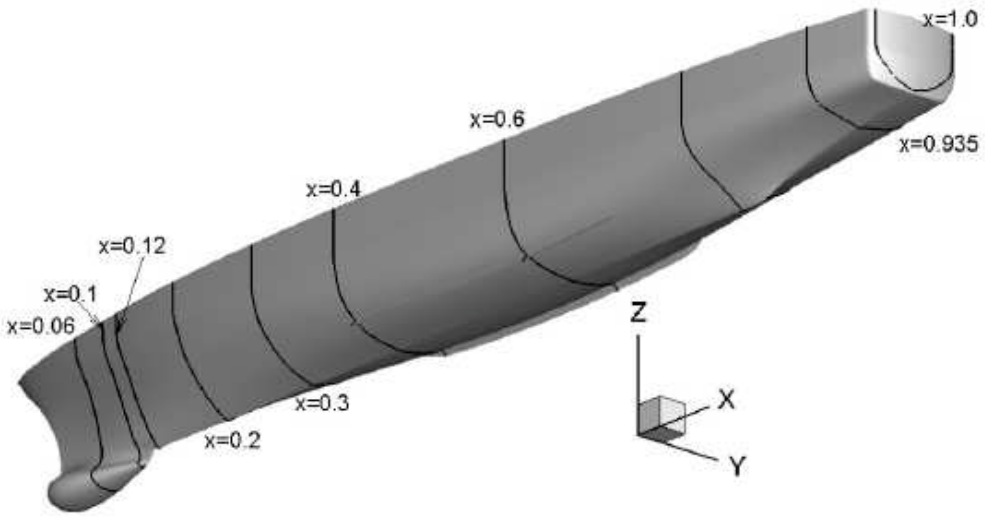
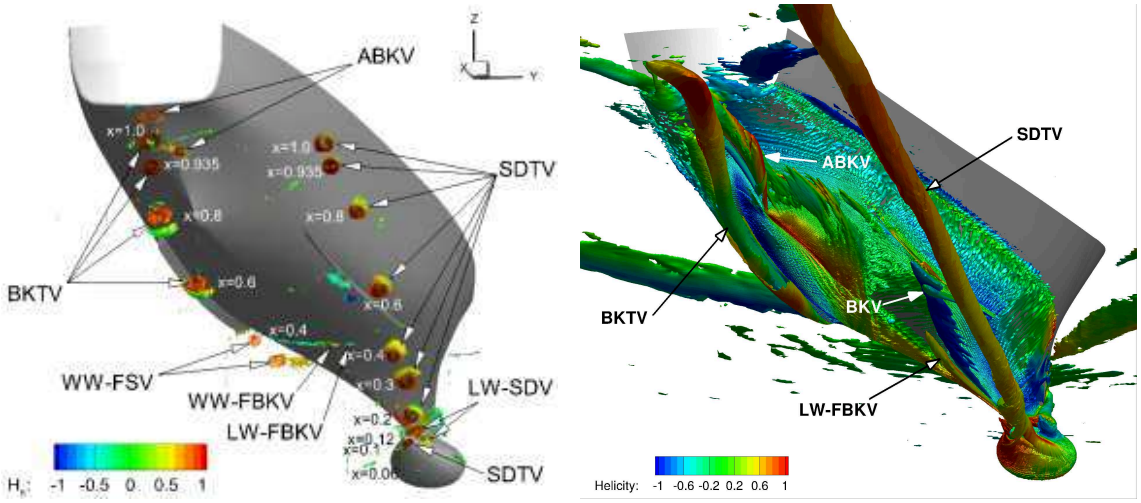


Figure 7.1 – Experimental measurement planes and coordinate system [116]

The flow topology was accurately described by Yoon et al. According to their way of naming the vortices, those originating from the port side are called windward (WW), while the vortices originating from the starboard side are called leeward (LW). Figure 7.2 shows an overview of the vortical structures in the TPIV IIHR measurements and the ISIS-CFD numerical simulation, with the help of the iso-surface for the non-dimensional $Q^* = 50$ colored by the helicity. One of the main longitudinal vortices, which originates from the windward surface of the sonar dome and is convected towards the stern with a clockwise rotation, is referred to as the sonar dome tip vortex (SDTV). The second, which is formed by cross-flow over the windward bilge keel, is called bilge keel tip vortex (BKTV). The intensity of the BKTV is comparable with the SDTV. Another vortex appears slightly behind the section $X/L = 0.8$ from the aft-body keel (ABKV). Other minor structures are also visible in Figure 7.2: the LW-SDV (leeward sonar dome vortex), the WW-FBKV and the LW-FBKV (windward and leeward fore body keel vortex) and the BKV (bilge keel vortex) which is formed because of the cross-flow over the leeward bilge keel. In this study, our attention is focused on the analysis of the numerical results relative to the main SDTV and BKTV vortices.



(a) TPIV measurements (b) Hybrid RANS/LES simulation

Figure 7.2 – DTMB 5512 at 10° static drift. Time-averaged $Q^* = 50$ surfaces colored by helicity for TPIV measurements [116] and hybrid RANS/LES simulation [104]

7.2 Numerical settings

An initial grid for the DTMB 5512 is generated with HEXPRESS™. It was observed in [104] that a mesh aligned with the undisturbed flow was beneficial for the preservation of the longitudinal vortices. For that reason, the same domain setup is used here, with the ship rotated over $\beta = 10°$ with respect to the X-axis (however, all results reported in this chapter use hull-aligned axes). The computational domain runs from $1.5L$ in front of the bow to $2.5L$ behind the stern. The width of the domain is $4L$ and the height is $2L$, while the bottom is at $1.5L$ below the undisturbed water surface.

The main challenge for the adaptive grid refinement is to capture the trailing vortical structures. Therefore, to limit the difficulty of projecting the refined grids onto the ship surface, the choice was made to create a rather fine original grid on the hull. This original grid resolves all the details of the keels, the sonar dome, the sharp bow and the skeg. As a result, the number of cells is high at $15M$. However, no refinement boxes are used so there

is no initial refinement away from the hull. On this grid, the averaged value of $y^+ = u_t y/\nu$ is 0.129, resulting in a first layer dimensionless thickness of the order of 7×10^{-6}.

For the RANS simulation, the anisotropic EARSM turbulence model [31] is used, while DDES-SST is selected as hybrid RANS-LES model. The reason for this choice is that in earlier tests on fixed grids, instabilities were observed when IDDES was applied to this case. On the other hand, the standard DES-SST model suffers from MSD due to the fine meshes [104]. Thus, DDES-SST is adopted as a compromise.

Following the earlier computations on fixed meshes, a time step of $\Delta t = 2 \times 10^{-2}$ is chosen for the RANS computation, while the DDES simulation uses $\Delta t = 5 \times 10^{-4}$ which is less than the Taylor time scale of $\Delta t_\lambda = 1.53 \times 10^{-3}$. A non-dimensional time scale, based on the time during which a particle goes from the tip of the sonar dome to the stern, is defined for the two simulations as $t \cdot (L/u)^{-1}$. The RANS simulation ran for 2700 time steps (27 non-dimensional units) while the DDES simulation was started from the final RANS result and continued for another 10,000 time steps (2.5 non-dimensional units). For visualization, the average solutions for DDES are computed over the last 4800 time steps of the simulation period (1.2 units).

7.3 Adaptive refinement strategy

The main choices for the adaptive grid refinement concern the criterion, the threshold, and the minimum cell size (Section 3.3), as well as the averaging strategy. For the refinement criterion, an unexpected benefit of the strategy to average the instantaneous criteria (Section 6.3.3) is that the software code which was developed to compute instantaneous-flow based criteria, could be used directly for the evaluation of the criteria to be averaged. This made it possible to quickly implement criterion averaging for most existing criteria. As a result, for the DTMB 5512 test case which is a free-surface flow, a combined free-surface and flux-component Hessian criterion is selected [107]. This criterion is based on the maximum of the FCH criterion and the free-surface capturing criterion (Section 3.3.2). The instantaneous version of this criterion was shown to be effective for the simulation of the DTMB 5415 in straight-ahead condition [82].

For the RANS simulations, instantaneous refinement criteria are normally used. However, for complex flows on fine meshes, some unsteadiness is sometimes detected, which may perturb the mesh adaptation. Thus, to get a mesh that is as smooth as possible, the RANS and the DDES meshes are both based on averaged refinement criteria.

For the threshold, standard ISIS-CFD guidelines suggest $T_r = L/1000$ for the free-surface part of the criterion, while for slender ships like the DTMB 5512 a fine RANS mesh is obtained when the Hessian threshold is set to $L/20$. These values are adopted for the RANS simulation. For DDES, since the resolved flow contains more small-scale structures, the refinement criterion will be higher than for RANS, even if it is averaged. Thus, for the same threshold, the refined mesh will be finer. A lower Hessian threshold could have been selected for the DDES-SST simulations. However, since it is considered interesting to see how much finer the mesh will become, the same threshold values as for RANS are chosen for the DDES simulation. Their effect on the number of cells is discussed later in Section 7.6.

In the fixed-mesh simulations of Visonneau et al. [104], great care was taken when selecting the cell sizes in the refinement boxes which capture the vortices emanating from the sonar dome and the bilge keel. The turbulent length scales were estimated at $\lambda = 2.34mm$ for the Taylor micro-scale and $\eta = 0.006mm$ for the Kolmogorov scale. To capture the larger eddies with at least four grid points, a cell size of $\lambda/4$ was used in the box from the sonar dome to $X/L = 0.4$ while $\lambda/2$ was chosen from $X/L = 0.4$ to the stern

at $X/L = 1$. To allow similar cell sizes in the adapted meshes, the minimum cell size is chosen as $\lambda/4 = 0.6mm$ for the current simulations.

The final choice concerns the averaging strategy, which depends on the frequencies and the physical behavior of the unsteady flow structures. Contrary to the backward-facing step of Chapter 5 which has the slow flapping-mode oscillation of its recirculation zone, the DTMB 5512 flow is thought to be globally steady. This means that the main unsteady features have relatively high frequencies, so they can be filtered out with a short sliding window. Furthermore, the trailing vortices are highly elongated structures and it takes a lot of time for the flow to pass from the beginning to the end of a vortex. This implies that a reactive refinement criterion is an advantage to adapt the grid everywhere in the vortex to the developing turbulent flow, as soon as possible.

Both these arguments favor a short sliding window. Therefore, a DDES averaging window of 250 time steps (0.0625 non-dimensional units) is chosen. With the mesh refined every 50 time steps, this means that 5 sets of criteria tables have to be saved, which is a reasonable quantity. For the RANS computation with its larger time steps, the mesh is refined every 25 time steps and the sliding window is set to 125 time steps (1.25 units) to keep the same number of tables. Finally, in the last part of the DDES simulation after the convergence is obtained, the mesh is refined every 100 time steps and the criterion averaging is applied over the entire computational time, to stabilize the mesh.

7.4 Global flow analysis

With these settings, EARSM and DDES-SST simulations using average-based AGR are performed. The final adapted mesh for EARSM has 47M cells and number of cells in the converged DDES mesh is about 153M. In this section, the obtained results of these simulation are compared with the IIHR experiments [116] from a global and local viewpoint.

7.4.1 Forces

Table 7.1 shows the axial force, lateral force and yaw moment corresponding to the EARSM and the DDES-SST simulations using average-based adaptive grid refinement, compared with the same simulations using a fixed grid with refinement boxes (163M cells). Using AGR in the simulation with the nonlinear anisotropic EARSM model, the errors between the predicted axial and lateral forces and the experiments are reduced by 1.4% and 1% respectively. For the yaw moment however, this error slightly increases. In the DDES-SST simulations, AGR does not improve the under-predicted drag coefficient and for the other forces, the discrepancies with the IIHR experimental data increase with respect to the fixed-grid results. Globally, large errors are observed in the prediction of the forces for both DES simulations of the DTMB 5512, which was also the case for the JBC [103].

Table 7.1 – Force coefficients on the DTMB 5512: axial force C_x, lateral force C_y, and yaw moment C_N

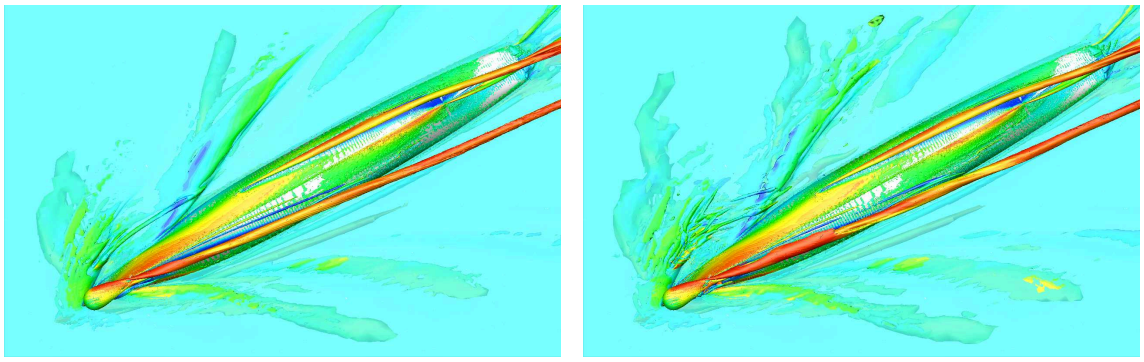
Case	Exp.	EARSM Fixed	EARSM AGR	DDES-SST Fixed	DDES-SST AGR
C_x	-19.61	-19.92 (-1.6%)	-19.58 (+0.2%)	-18.20 (+7.2%)	-18.19 (+7.2%)
C_y	58.46	62.25 (+6.5%)	61.67 (+5.5%)	62.72 (+7.3%)	65.57 (+12.2%)
C_N	28.61	29.17 (+2.0%)	29.36 (+2.6%)	29.30 (+2.4%)	29.90 (+4.5%)

The reason for the large difference with the experimental data in the two DDES computations is probably that a fine grid in the wall-parallel direction is needed where

the vortices lie near the wall, in order to resolve the vortex cores. These regions have about the same grid size for the fixed-grid and adaptively refined mesh (see Section 7.6 for more discussion on the refined mesh). In these regions, the mesh forces the DDES model to provide a wall modelling in LES (WMLES, see Chapter 2), which results in the log-layer mismatch reported by Nikitin et al. [72]. Additional computations based on the IDDES variants can be carried out to check this hypothesis, since the WMLES branch of the IDDES model could be activated by the presence of the locally fine grids and LES turbulent content which comes from vortical structures, thus resolving this part of the boundary layers more accurately.

7.4.2 Global flow field

Figures 7.3 and 7.4 provide a bottom view of the main time-averaged and instantaneous vortical structures created around the DTMB 5512 at 10° static drift. These figures show the iso-surface $Q^* = 50$ colored by the helicity as computed by the EARSM and DDES-SST models. In the time-averaged views, the three main vortices are observed which are more or less similar in the two simulations in terms of location and longitudinal extent. The SDTV arises from the sonar dome and is convected to the stern of the ship. DDES-SST provides a SDTV vortex with a radius which is smaller all along its progression compared to EARSM, a behavior that was observed by Visonneau et al. [104] as well. In addition, the BKTV and the ABKV are seen in both computations. Like the SDTV, the radius of these vortices for DDES-SST is smaller than for RANS. This indicates a more intense longitudinal vorticity for the DDES-SST computation.

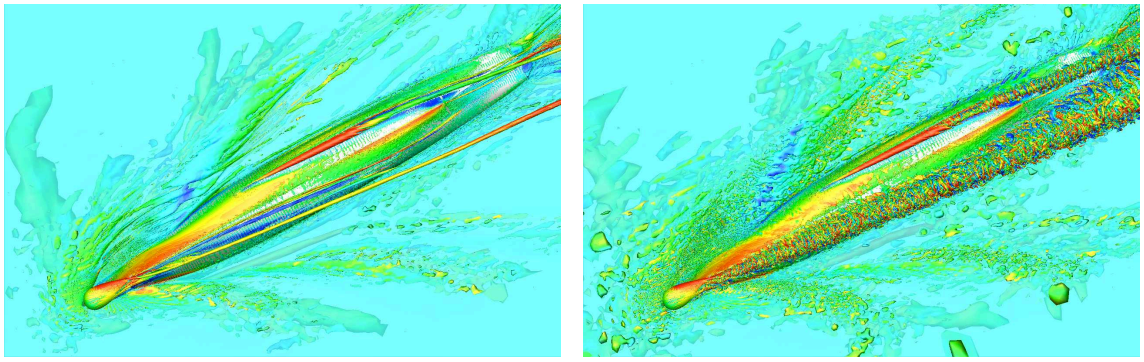


(a) Time-averaged (b) Instantaneous

Figure 7.3 – EARSM. Global view of the time-averaged and instantaneous $Q^* = 50$ surfaces colored by the helicity

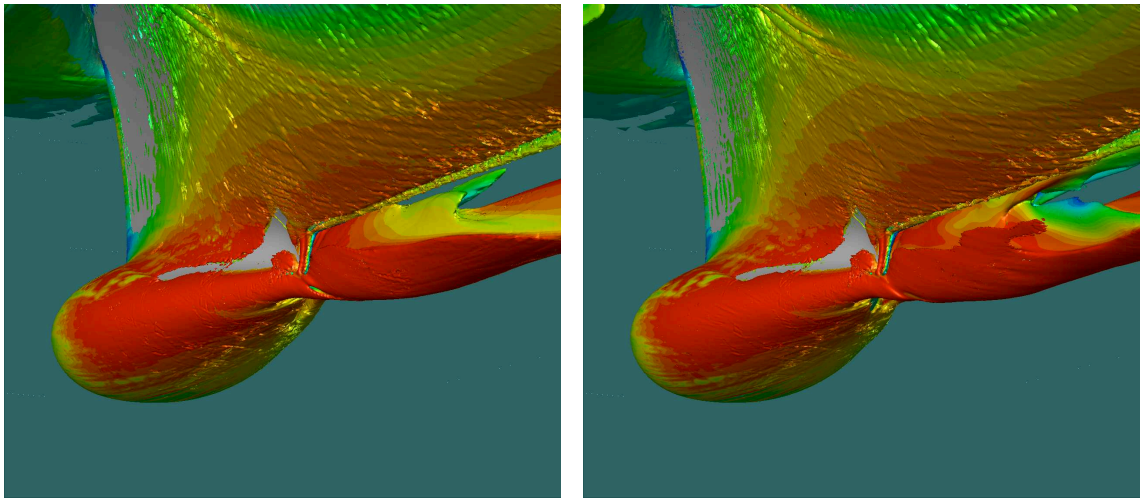
Figures 7.5 to 7.8 give a closer view of the vortices' onset at the sonar dome on the windward and leeward sides. The SDTV detaches from the tip of the windward side of the sonar dome, slightly before the vertical trailing edge for the EARSM and DDES-SST computations. Once the SDTV enters the wake of the trailing edge, the separation which takes place along the keel line interacts with the SDTV (Figures 7.5 and 7.6).

On the leeward side, several more axial vortices are observed for the two cases (Figures 7.7 and 7.8). The first structure is a small closed recirculation zone of triangular shape located at the intersection of the sonar dome and the stem. The reattachment of this structure on the leeward surface of the hull is observed in the EARSM and the DDES-SST computations. Below the recirculation zone, a counter-rotating longitudinal separation (with blue color) detaches from the upper part of the sonar dome for both the EARSM and DDES-SST cases (Figures 7.7a and 7.8a). This separation is referred to as



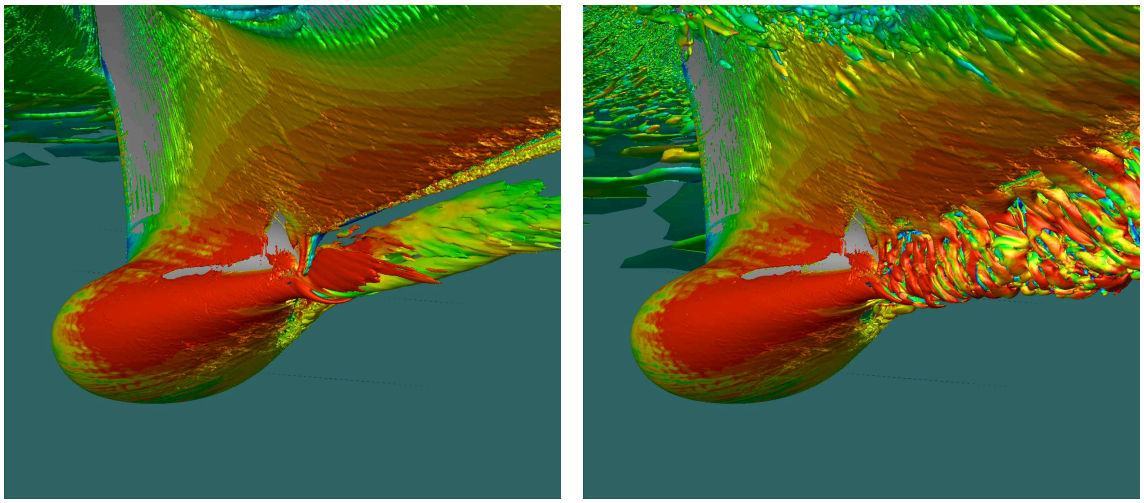
(a) Time-averaged (b) Instantaneous

Figure 7.4 – DDES-SST. Global view of the time-averaged and instantaneous $Q^* = 50$ surfaces colored by the helicity



(a) Time-averaged (b) Instantaneous

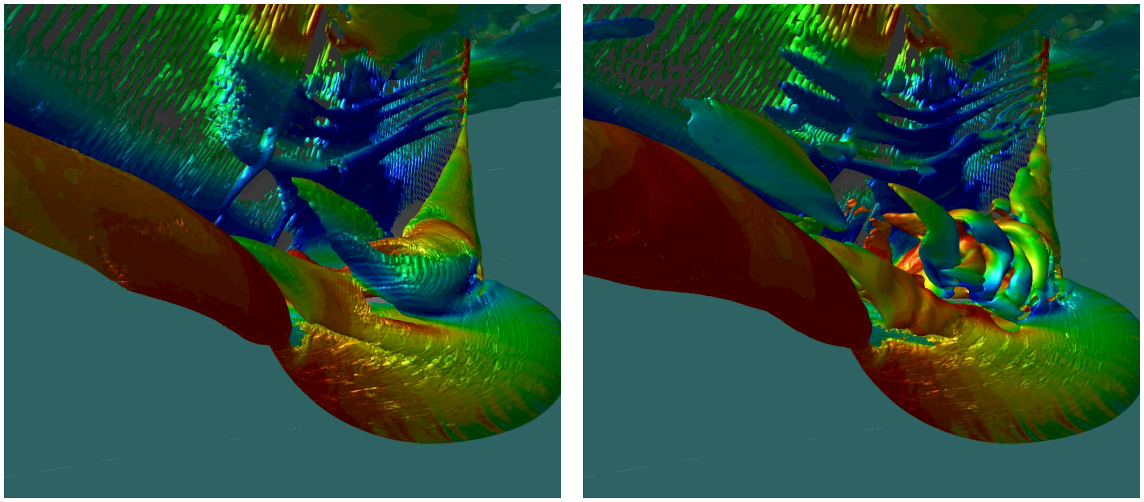
Figure 7.5 – EARSM. Iso-surfaces of the time-averaged and instantaneous $Q^* = 50$ on the windward side of the sonar dome



(a) Time-averaged (b) Instantaneous

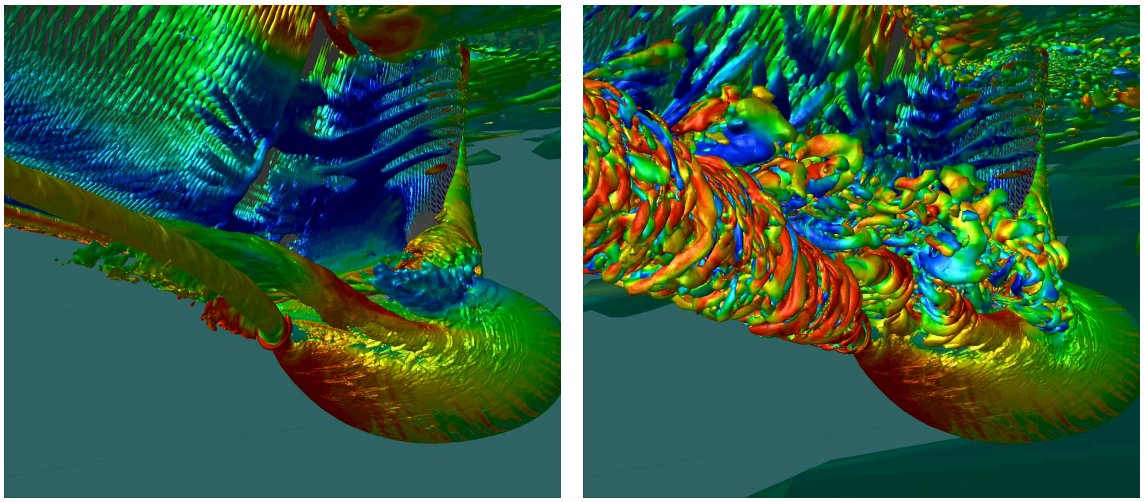
Figure 7.6 – DDES-SST. Iso-surfaces of the time-averaged and instantaneous $Q^* = 50$ on the windward side of the sonar dome

the Leeward Sonar Dome Bubble. The third leeward longitudinal vortex emerges from the middle part of the sonar dome. This vortex was identified by Yoon et al. [116] as Leeward Sonar Dome Vortex (LW-SDV) and also detected by Visonneau et al. [104], independently of the turbulence closure. It merges with the main SDTV.



(a) Time-averaged (b) Instantaneous

Figure 7.7 – EARSM. Iso-surfaces of the time-averaged and instantaneous $Q^* = 50$ on the leeward side of the sonar dome



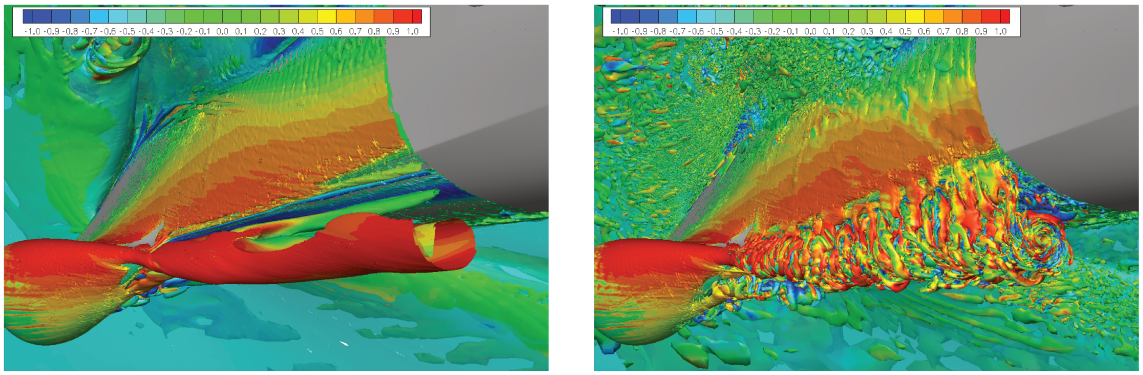
(a) Time-averaged (b) Instantaneous

Figure 7.8 – DDES-SST. Iso-surfaces of the time-averaged and instantaneous $Q^* = 50$ on the leeward side of the sonar dome

Both side views of the instantaneous DDES-SST solution (Figures 7.6b and 7.8b) show that this solution is highly unsteady. Contrary to the averaged view, the instantaneous flow is composed of small unsteady vortices. These deforming unsteady structures, created by the periodic shedding of ring vortices of various helicities, are convected along the hull in the vicinity of the sonar dome and are finally absorbed into the SDTV vortex. In Figure 7.9, a cross-section of the instantaneous $Q^* = 50$ iso-surface for the SDTV vortex at $X/L = 0.3$ is presented. The figure shows that for the DDES-SST simulation, the multitude of intense ring-like vortices coming from the periodic shedding at the trailing edge of the sonar dome

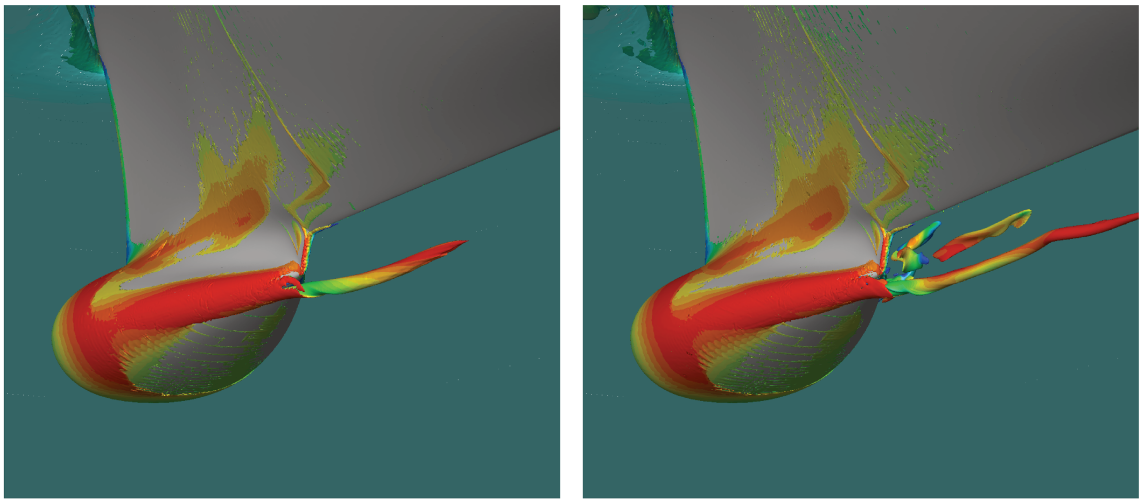
and from the unsteady separation present on the leeward side, are present throughout the SDTV core. As a result of irregular rotations and frequent collisions of these small structures with the small-diameter cylinder-shaped iso-surface in the center of the SDTV, a slight meandering is imposed to this vortex core center. The effect of this meandering motion of the vortex center is discussed in the following section.

Unlike the fixed-grid computation, some unsteadiness is also detected in the solutions of the EARSM computation using AGR (Figure 7.7), particularly related to the counter-rotating separation on the leeward side of the sonar dome. This unsteadiness is contrary to the nature of RANS computations (Section 2.1). For a higher value of $Q^* = 10,000$, comparing the instantaneous and time-averaged EARSM simulation in Figure 7.10 shows that the mean iso-surface of the SDTV is much shorter than the instantaneous one. This is probably an indication of a vortex meandering motion, but a much weaker one than what is induced by the ring-like vortices observed in the DDES simulation (Figure 7.8b). The unsteady vortices on the leeward side of the sonar dome in the EARSM simulation likely perturb the main vortex core and are the cause of its slight movement.



(a) EARSM (b) DDES-SST

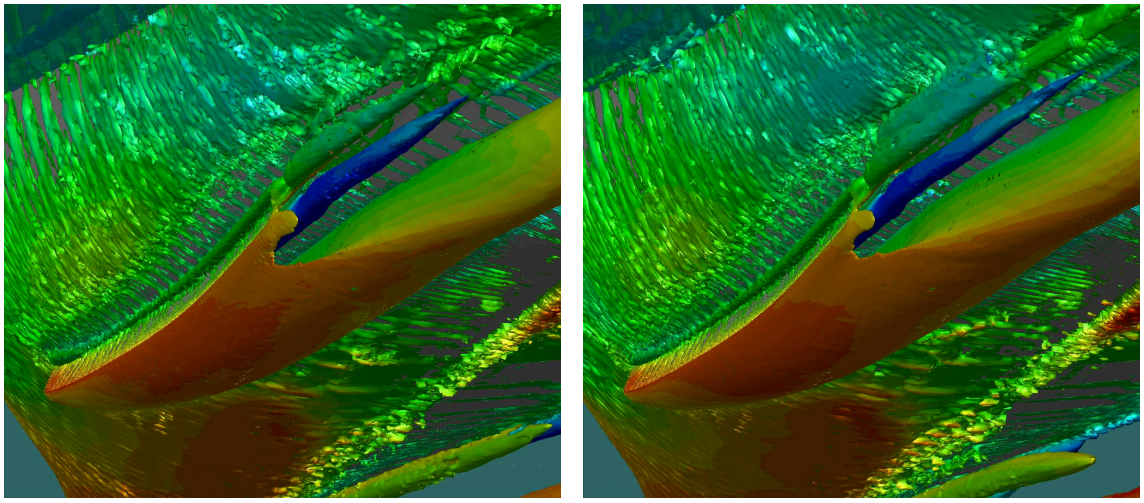
Figure 7.9 – Cross-section of the instantaneous $Q^* = 50$ isosurface for the SDTV vortex at $X/L = 0.3$



(a) Time-averaged (b) Instantaneous

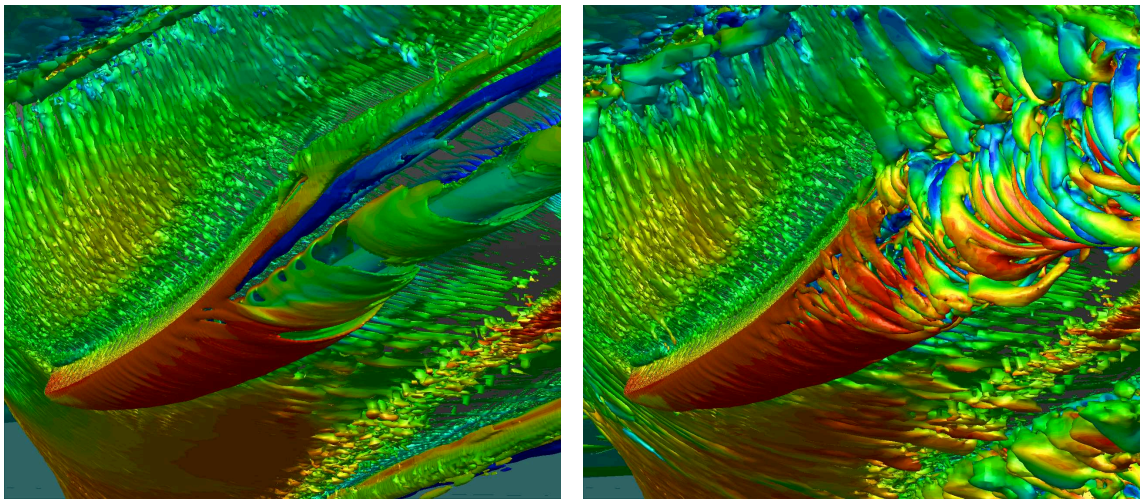
Figure 7.10 – EARSM. Iso-surfaces of the time-averaged and instantaneous $Q^* = 10,000$ on the windward side of the sonar dome

For comparison with the SDTV, Figures 7.11 and 7.12 show the creation of the BKTV. For both the EARSM and DDES-SST computations, the BKTV originates from the tip of the windward bilge keel. In addition, due to the cross-flow over the bilge keel, a more or less stable shear layer is created. For the DDES simulation, destabilization of the shear layer is observed starting from the aft part of the bilge keel. This produces unsteadiness which later on is absorbed into the BKTV. As for the SDTV, the unsteady ring-like vortices coming from the shear layer around the vortex core cause a slight movement of the BKTV. In addition, for both the simulations, a counter-rotating secondary vortex at the back of the bilge keel is seen. For DDES, this vortex is unsteady.



(a) Time-averaged (b) Instantaneous

Figure 7.11 – EARSM. Iso-surfaces of the time-averaged and instantaneous $Q^* = 50$ around the windward bilge keel



(a) Time-averaged (b) Instantaneous

Figure 7.12 – DDES-SST. Iso-surfaces of the time-averaged and instantaneous $Q^* = 50$ around the windward bilge keel

It is reassuring to notice that using AGR, the same number of vortices and the same locations of onset in an averaged sense are observed for the EARSM and hybrid RANS-LES

turbulence models, compared with the towing tank experiments and also the simulations with a fixed mesh.

7.4.3 Cross-sections

In Figures 7.13 to 7.16, the solutions of the EARSM and DDES-SST computations at several cross-sections are compared with the IIHR measurements of the longitudinal velocity field, the longitudinal vorticity and the turbulence kinetic energy. The evolution of the local grid refinement around the core of the main axial vortices is shown as well.

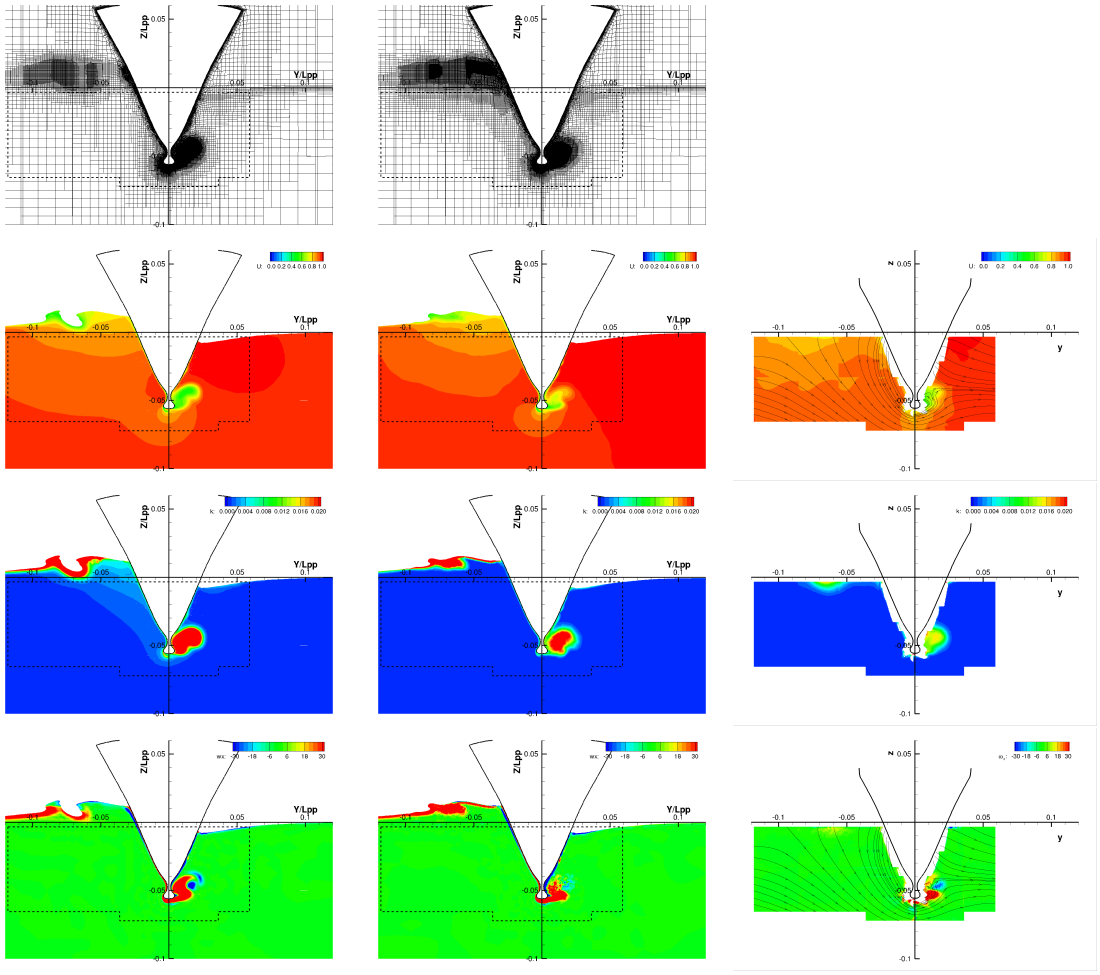


Figure 7.13 – $X/L = 0.1$. Top to bottom: adapted mesh, averaged axial velocity, TKE and axial vorticity for EARSM (left), DDES-SST (middle) and Experiments (right)

The cross-section $X/L = 0.1$ (Figure 7.13) corresponds to the trailing edge of the sonar dome where the development of the longitudinal vortical structures on the leeward side occurs (Section 7.4.2). The density of the fine grids on the leeward side of the sonar dome is due to the local unsteady interacting structures with a high level of turbulence kinetic energy predicted by both the EARSM and DDES-SST turbulence closures. The level changes of the free surface on both sides of the hull and the breaking waves as a result of the hull drift are also captured by the AGR.

In both the simulations, the intensity of the turbulence kinetic energy (TKE) is over-predicted compared to the experimental measurements. For the EARSM, the region with

a high TKE value is even larger than for DDES-SST. In addition, the red spot for ω_x associated with the main SDTV vortex, which is about to be established, is accompanied by a counter-rotating vortex (blue spot) captured most clearly by the EARSM computation, but visible in all results. This is probably the Leeward Sonar Dome Bubble.

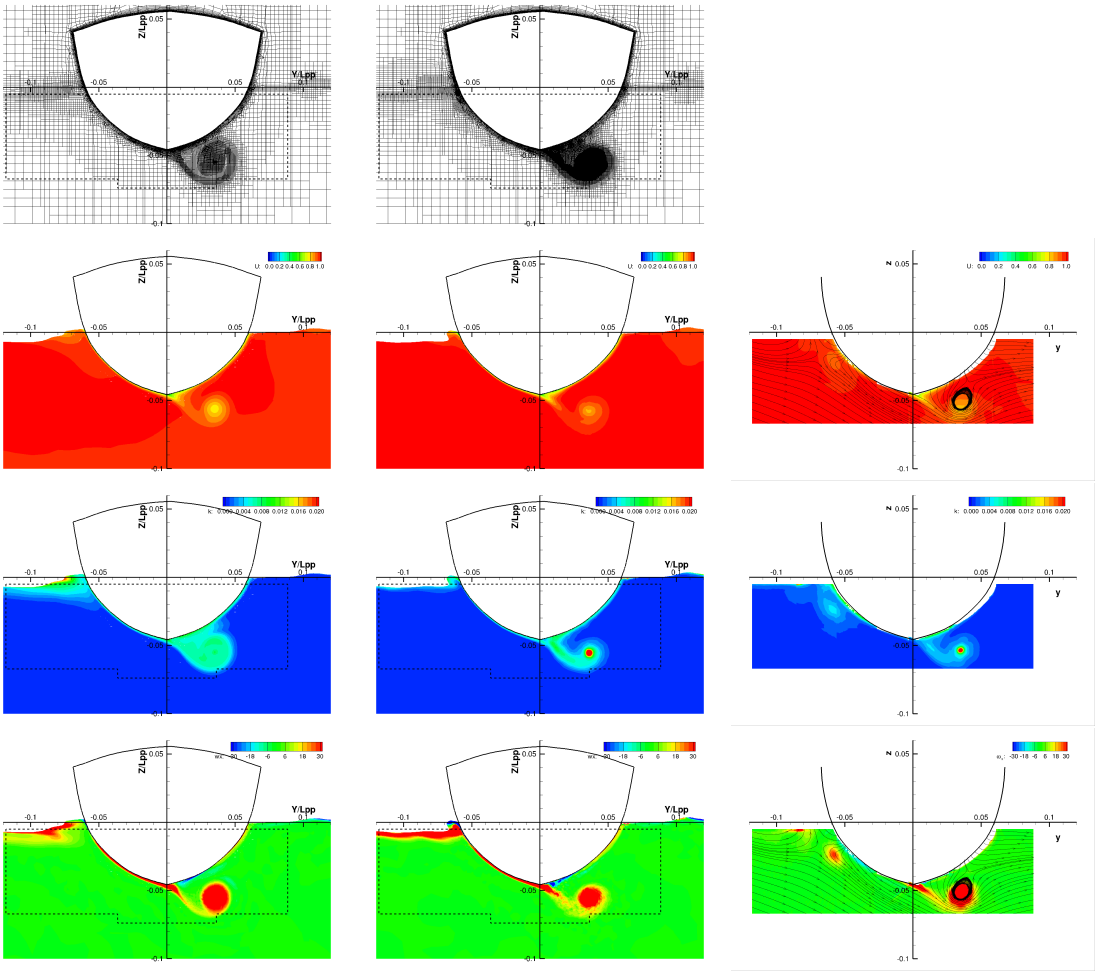


Figure 7.14 – $X/L = 0.3$. Top to bottom: adapted mesh, averaged axial velocity, TKE and axial vorticity for EARSM (left), DDES-SST (middle) and Experiments (right)

At cross-section $X/L = 0.3$ in Figure 7.14, for the DDES-SST simulation, it is seen that the refinement criterion captures the highly unsteady structures shed from the trailing edge and the leeward side of the sonar dome into the core of the SDTV, because it refines locally the grids in the core. In the adapted mesh, the local grids in the core are refined up to the minimum cell size selected, and a more or less isotropic mesh is obtained in this region. For the EARSM computation in contrast, just a small spot in the center of the vortex has a grid with the minimum cell size. For both simulations, the shear layer of the separation from the keel line is also captured by the AGR.

The TKE at the center of the SDTV for the two simulations has a maximum value. While the EARSM turbulence model usually minimizes the TKE of the core vortex [104], in the RANS simulation using AGR, a weak maximum appears (Figure 7.14). This maximum TKE value for EARSM can be the result of the unsteady counter-rotating vortices already described in Section 7.4.2 which cause a slight meandering of the main SDTV core. This small movement creates fluctuations in the velocity field and thus, the TKE value increases

in the region.

In the DDES-SST simulation, the presence of unsteady ring vortices causes the production of resolved turbulence in the outer vortex and the shear layer from the keel line. However, the core of the SDTV is different. As seen in the instantaneous iso-surface $Q^* = 50$ at the cross-section $X/L = 0.3$ for the DDES-SST simulation (Figure 7.9b), the unsteady turbulent structures do not exist in a small-diameter vortex core in the center of SDTV, where the TKE value is at its maximum. The small movements of the vortex core as a result of these structures' collisions with the core create high-level velocity fluctuations which lead to elevated values of turbulence kinetic energy at the center of the vortex, that are observed in the measurements as well. The higher TKE at the vortex core for DDES-SST with respect to EARSM is due to the ring vortices around the core that probably cause stronger vortex meandering.

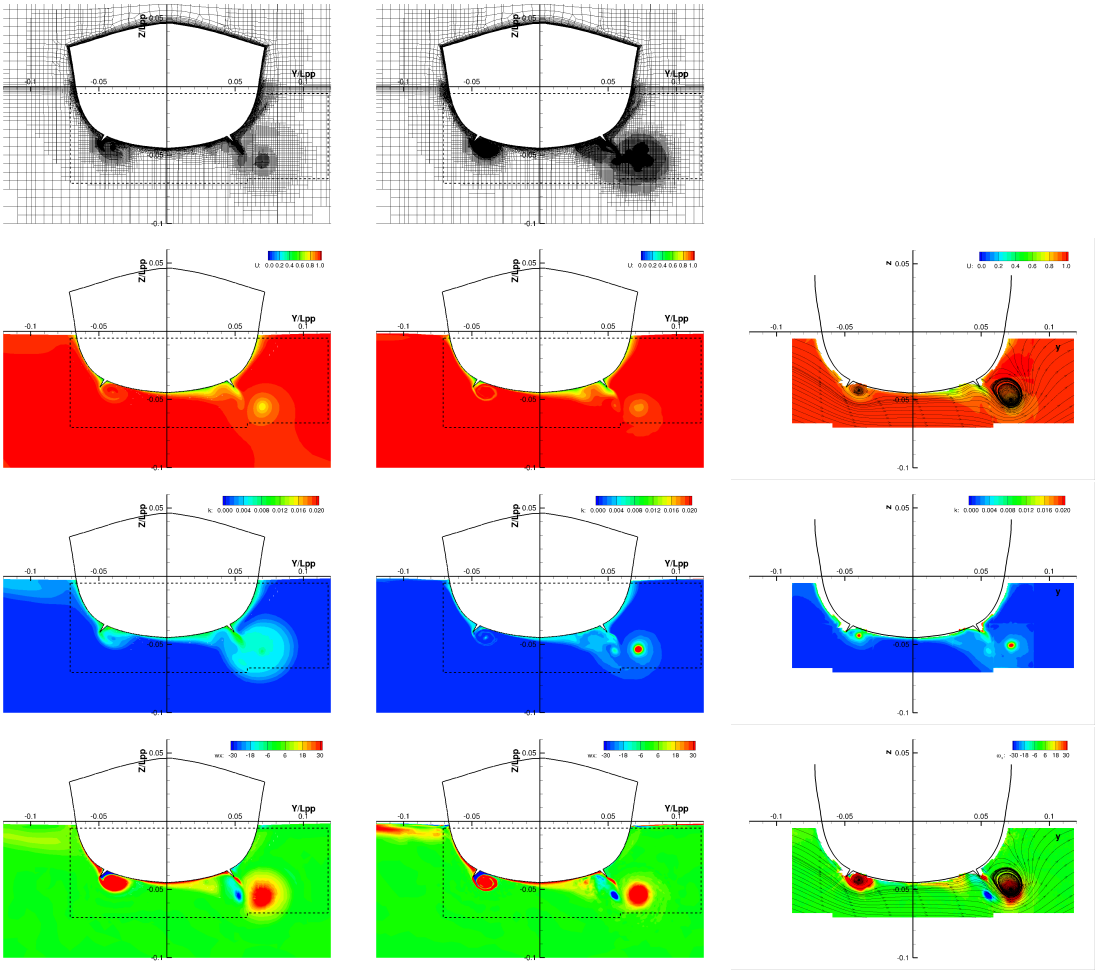


Figure 7.15 – $X/L = 0.6$. Top to bottom: adapted mesh, averaged axial velocity, TKE and axial vorticity for EARSM (left), DDES-SST (middle) and Experiments (right)

Figure 7.15 at $X/L = 0.6$ shows the onset of the BKTV behind the windward bilge keel. The steady shear layer over the windward bilge keel (Figures 7.11 and 7.12) with strong second derivatives creates refinement around the bilge keel. Like in the core of the SDTV, the local grid size for the DDES-SST is equal to the minimum cell size and for the EASRM simulation, the grids are a bit coarser. In addition, the structures coming from the sonar dome and the keel line, combined with the wake of the leeward bilge keel, create

a shear layer that is captured by the refinement criterion. In a global bottom view for the DDES-SST simulation (Fig 7.4b), it is seen that these turbulent features are preserved up to the stern.

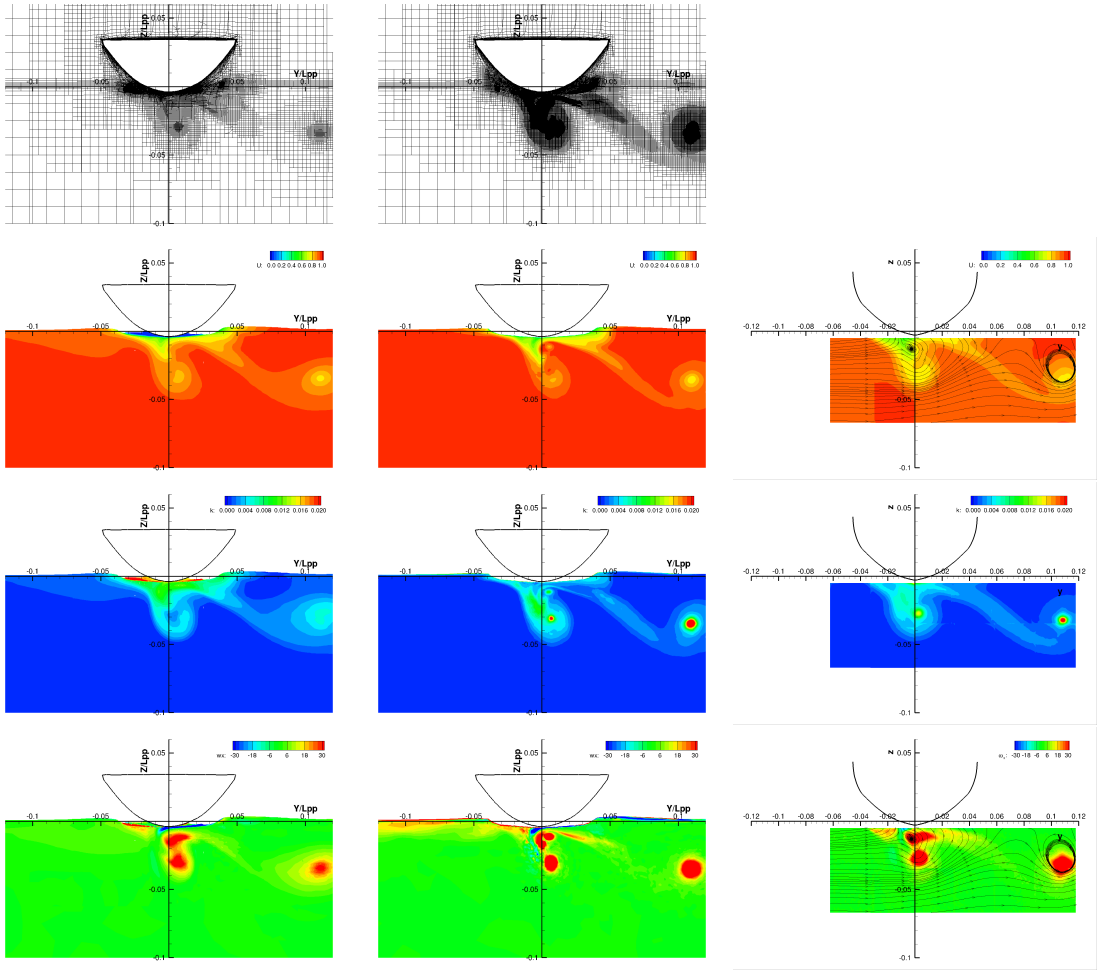


Figure 7.16 – $X/L = 1$. Top to bottom: adapted mesh, averaged axial velocity, TKE and axial vorticity for EARSM (left), DDES-SST (middle) and Experiments (right)

The progression of the SDTV along the hull is accompanied by a high intensity of TKE in its core for the DDES-SST simulation. However, the TKE value is underpredicted at the center of the BKTV compared to the experimental data. This can be caused by the effect of the RANS zone in the boundary layer around the bilge keel which implies that the flow approaching the bilge keel does not contain unsteadiness. Unlike the SDTV, which absorbs the detached turbulent features from the sonar dome surface that force the model to resolve the turbulence, the stable shear layer over the bilge keel cannot create resolved TKE fast enough. In addition, the lack of turbulent instabilities implies that the core of the BKTV does not have the small movements which the SDTV undergoes and that therefore, the velocity fluctuations which lead to high TKE at the core of the vortex are not created. Thus, the predicted TKE are much lower than in the experiments.

In the EARSM simulation however, the low TKE value at the center of the BKTV corresponds to the natural behavior of this closure in steady vortices. For the SDTV, the unsteady effects of the upstream structures are diminished along the hull by this statistical closure. This reduces the maximum TKE value at the center of the SDTV. In addition, the

axial velocity in the core of the SDTV is under-predicted by the EARSM-based simulation compared to the DDES-SST and the experimental data.

At the back of the ship, at cross-section $X/L = 1$ (Figure 7.16), in addition to the mesh adaptation around the SDTV, the BKTV and the separated shear layer from the leeward keel, AGR detects another region with locally high second derivatives, which corresponds to the aft-body keel vortex . For the DDES-SST simulation, the highly unsteady solutions result in smaller cells around the vortex cores than for the EARSM simulation, in which the grid size becomes coarser due to the high dissipation rate of the turbulence along the hull.

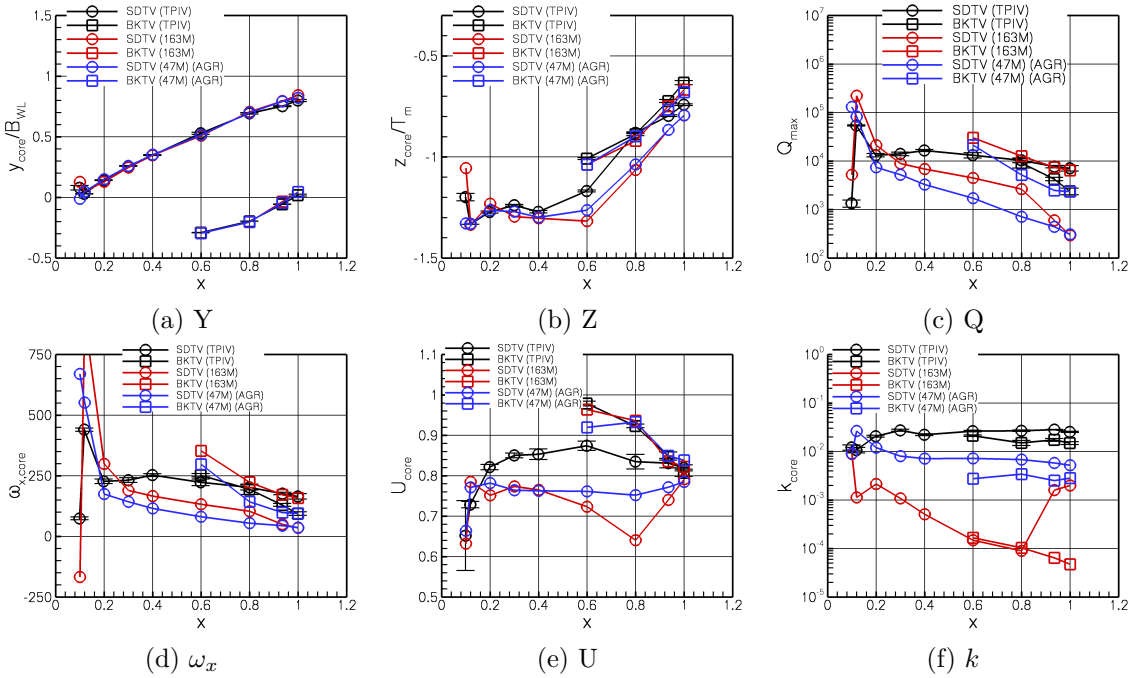
In the DDES-SST simulation, the underestimated TKE value in the core of the BKTV at $X/L = 0.6$ is replaced by a high-TKE vortex core. The destabilization of the windward bilge keel shear layer (Figures 7.4b and 7.12b) creates unsteady structures from the end of the bilge keel which are advected into the BKTV. As for the SDTV, the vortex meandering due to the presence of these structures and therefore the creation of velocity fluctuations in the center of the BKTV, is the reason for the increase in turbulence kinetic energy at $X/L = 1$. However, for the EARSM computation, the shear layer coming from the bilge keel remains stable (Figure 7.11b) and contrary to the SDTV, no unsteady solution is observed in this regions, that could have caused vortex meandering. Thus, as it is expected from the EARSM model, the TKE in the vortex core has a minimum value. Regarding the aft-body keel vortex (AFKV), the TKE is overpredicted in the vortex core for the DDES simulation compared to the experiments. For EARSM however, the TKE is underpredicted due to errors in the model.

7.5 Longitudinal evolution

For a more detailed comparison of the different simulations, the evolution of the flow in the vortex cores is studied. Figures 7.17 and 7.18 provide a comparison between the IIHR measurements at the core of the SDTV and BKTV vortices, the EARSM and DDES computations performed by Visonneau et al. [104] on a fixed grid comprised of 163M cells as a reference, and the present computations, performed with the same turbulence closures but using adaptive meshes. In all cases, the locations of the vortex cores are determined by the maximum value of the second invariant Q^*.

In the EARSM computation using AGR, the number of cells in the final adapted mesh is almost 70% less than in the simulation on the fixed mesh. The predicted lateral and vertical positions of the centers of the SDTV and BKTV are in a good agreement with the reference simulation and the experimental data (Figures 7.17a and 7.17b). When AGR is used, the discrepancies between the simulation and the experimental measurements of the second invariant Q^* (Figure 7.17c) and the longitudinal vorticity ω_x (Figure 7.17d) are increased, notably for the SDTV. The most remarkable impact of dynamically adapting the mesh instead of using a fixed mesh in the EARSM computation is the increase of the turbulence kinetic energy at the center of the vortices (Figure 7.17f). In the EARSM computation, it is expected that the model minimizes the TKE in the vortex cores which is the case for the fixed-grid simulation. However, the TKE intensity has a maximum in the cores on adapted grids, due to the unsteadiness in the turbulent structures which detach from the sonar dome surface. The unsteadiness also explains the reduction of Q^* and ω_x: if the vortex core moves, the mean solution is spread out and its peaks diminish. The increase in the prediction of the longitudinal velocity component U at the core of the SDTV between $X/L = 0.6$ and $X/L = 1$ (Figure 7.17e) happens for the same reason.

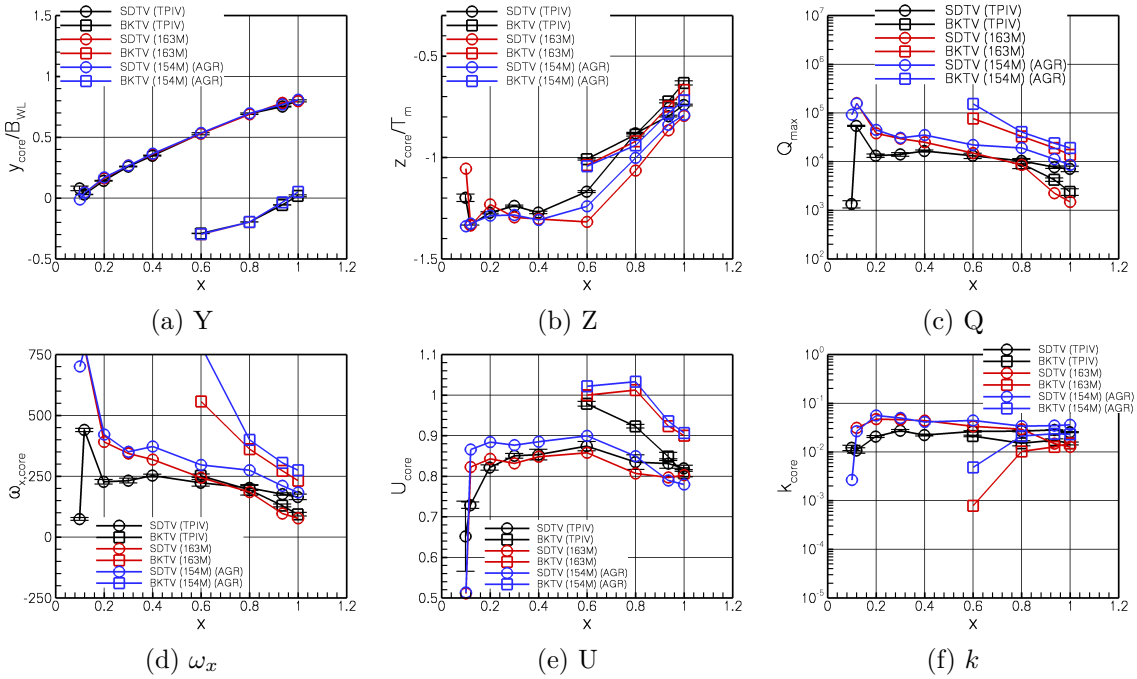
For the DDES-SST simulations with AGR, the final adapted mesh has a slightly lower number of cells (154M) than the simulation with a fixed mesh. As for the EARSM



(a) Y (b) Z (c) Q

(d) ω_x (e) U (f) k

Figure 7.17 – EARSM. Evolution of the Y- and Z-coordinates, the second invariant Q^*, the longitudinal vorticity ω_x and velocity U, and the turbulence kinetic energy k along the centers of the SDTV and BKTV vortex cores



(a) Y (b) Z (c) Q

(d) ω_x (e) U (f) k

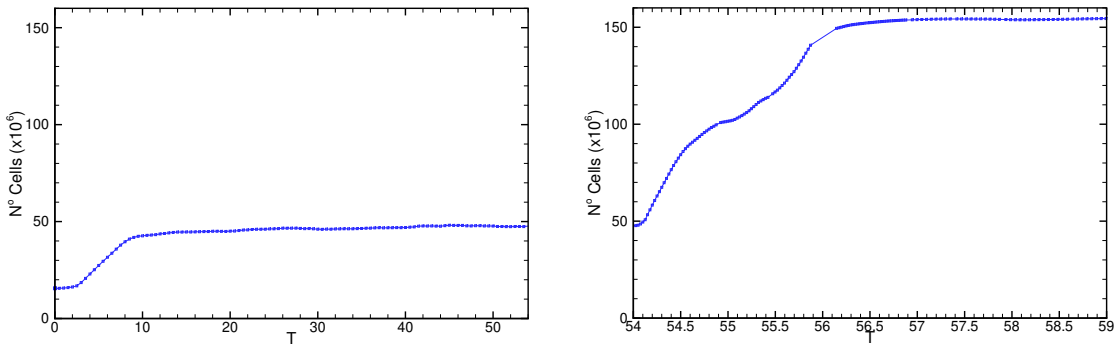
Figure 7.18 – DDES-SST. Evolution of the Y- and Z-coordinates, the second invariant Q^*, the longitudinal vorticity ω_x and velocity U, and the turbulence kinetic energy k along the centers of the SDTV and BKTV vortex cores

computation, the positions of the vortex cores are well predicted (Figure 7.18a and 7.18b). The second invariant Q^* (Figure 7.18c) and the longitudinal vorticity ω_x (Figure 7.18d) are predicted very close to the fixed-grid reference simulation up to $X/L = 0.4$. Beyond that however, Q^* and ω_x are underestimated on the reference grid. This is because in the reference simulation, the finest refinement box covers a region from the sonar dome to the cross-section $X/L = 0.4$, after which the size of the cells becomes two times coarser. In opposite, AGR still keeps the mesh refined around the vortex cores up to the stern. The axial velocity and the TKE intensity at the vortex cores are also well predicted with respect to the reference solution and the experimental data, with a higher prediction for regions where there is no fine refinement box on the fixed grid (e.g. for the BKTV, see Figure 7.18f).

In general, thanks to the average-based adaptation, the solution of the reference DDES-SST simulation is reproduced without user effort for the manual placement of the refinement boxes around the vortices. In addition, due to the optimal distribution of the fine grids in the entire domain, the refinements around the vortices are maintained up to the aft part of the ship with a lower number of cells. On the other hand, for the EARSM simulation, although the errors between the prediction and the measurements are reduced significantly, the observation of unsteadiness in the solution of this RANS computation brings up some doubts about the validity of this turbulence model for the current simulations.

7.6 Convergence of the mesh adaptation

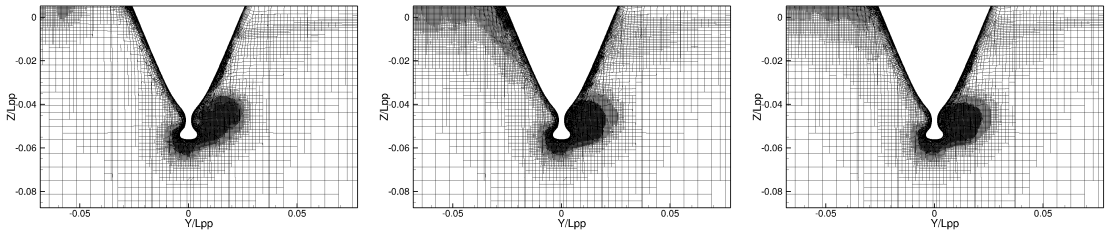
In this section, the evolution of the adapted mesh during the EARSM and DDES-SST simulations is investigated. Figure 7.19 shows the variation of the number of cells in the adaptively refined grids with respect to the computational time. In the RANS simulation, the number of cells increases steadily, since AGR is capturing the vortices as they slowly grow in length (Figure 7.19a). After almost 500 time steps (5 non-dimensional units), the rate of increase in the number of cells reduces, however the mesh does not really converge until 2500 time steps.



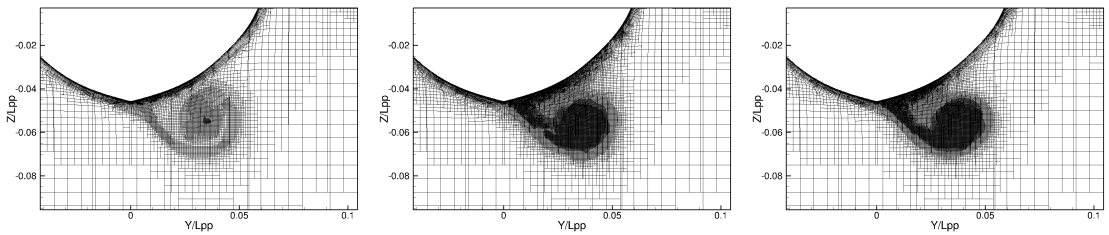
(a) EARSM (b) DDES-SST

Figure 7.19 – Evolution of the number of cells in the adaptively refined grids, for EARSM and DDES-SST (restarted from the EARSM solution)

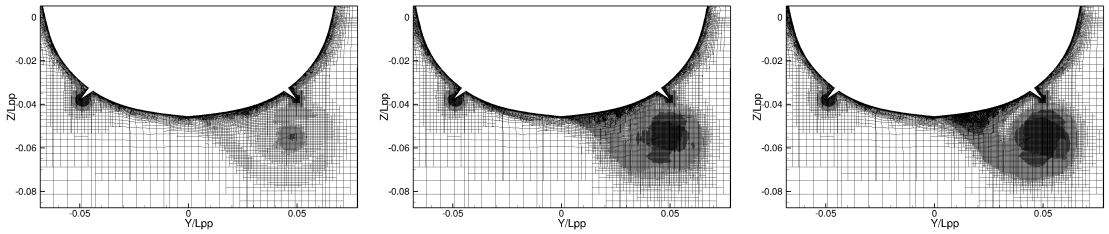
Then, the DDES-SST simulation starts from the final refined RANS mesh. Figure 7.19b shows how the number of cells increases for the same threshold as the EARSM simulation. The DDES resolved solution has more small-scale structures which create intense fluctuations. The instantaneous-based criteria computed from these highly unsteady solutions are higher than those for EARSM. This is the reason for the increase in the number of cells when the DDES solution is started on the converged mesh of the RANS simulation.



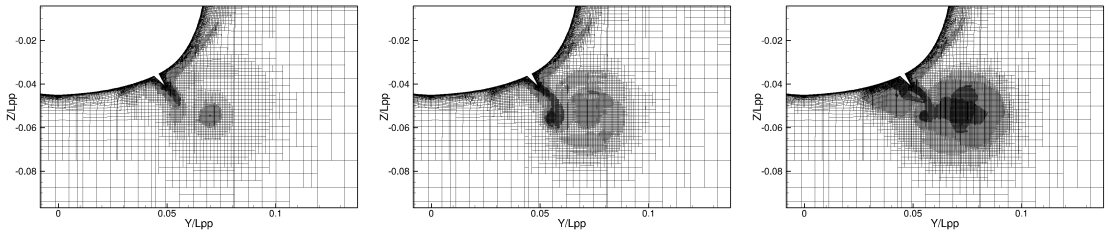
(a) $X/L = 0.1$



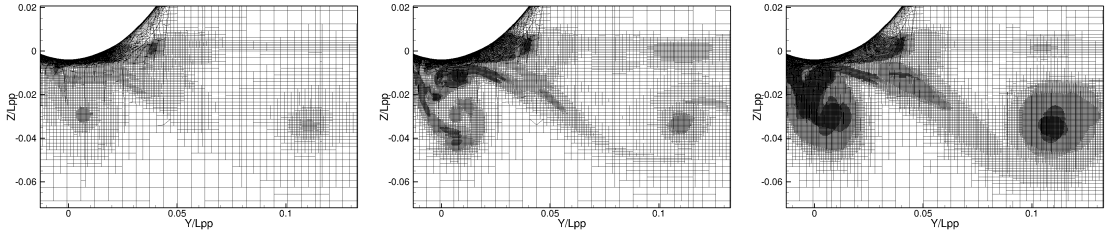
(b) $X/L = 0.3$



(c) $X/L = 0.4$



(d) $X/L = 0.6$



(e) $X/L = 1$

Figure 7.20 – The adapted mesh in DDES-SST simulation. At the beginning of the computation (left), after 2700 time steps (middle) and at the end of the computation (right)

However, compared with the EARSM RANS computations, the adapted mesh is converged much faster in the DDES simulation.

Figure 7.20 represents the adapted mesh of several cross-sections for three different instants. The first column on the left corresponds to the final mesh of the RANS computation and the beginning of the DDES-SST simulation. The second set of meshes shows the adapted mesh after the first 2700 time steps (0.675 non-dimensional units) and the last column is related to the end of the DDES-SST computation (10,000 time steps or 2.5 non-dimensional units). It is seen that after only 0.675 non-dimensional units the adapted meshes are almost converged for the cross-sections located up to $X/L = 0.4$ since, at the end of the simulation, no significant further refinement is observed. In addition, for the flow features which are originated from locations further aft than the sonar dome, the convergence of the local grids is also observed, like the wake of the leeward bilge keel at $X/L = 0.6$. This means that the mesh is refined almost as soon as the first resolved turbulence has moved from its point of creation on the hull to the given position. From this, it can be implied that in the DDES-SST simulation, the full mesh convergence should be obtained in a little more than 1 non-dimensional unit (4000 time steps or 2 seconds), which agrees well with Figure 7.19b.

To evaluate the choice of the 250 time step averaging window for the refinement criterion, the time-averaged velocity of a point probe, for several averaging intervals, is presented in Figure 7.21 for the DDES simulation. Increasing the averaging interval diminishes the chaotic behavior of the high-frequency flow features and leads to a smoother averaged solution, as it is already discussed in Section 5.4. A shorter interval leads to a reactive refinement criterion which adapts the mesh with respect to lower frequency flow structures, like the flapping motion of backward-facing flows. While this is useful to get rapid mesh convergence, it was intended in this test case to reach a converged mesh that does not undergo frequent changes due to the flow features. According to Figure 7.21, the averaged velocity still contains fluctuations even with the interval of 250 time steps. To average out the fluctuations, a longer averaging interval can be an option, but this requires more memory to save the solutions. Thus, starting with a short window and then switching to the averaging scheme over the whole computational time for the later part of the simulation is a logical choice.

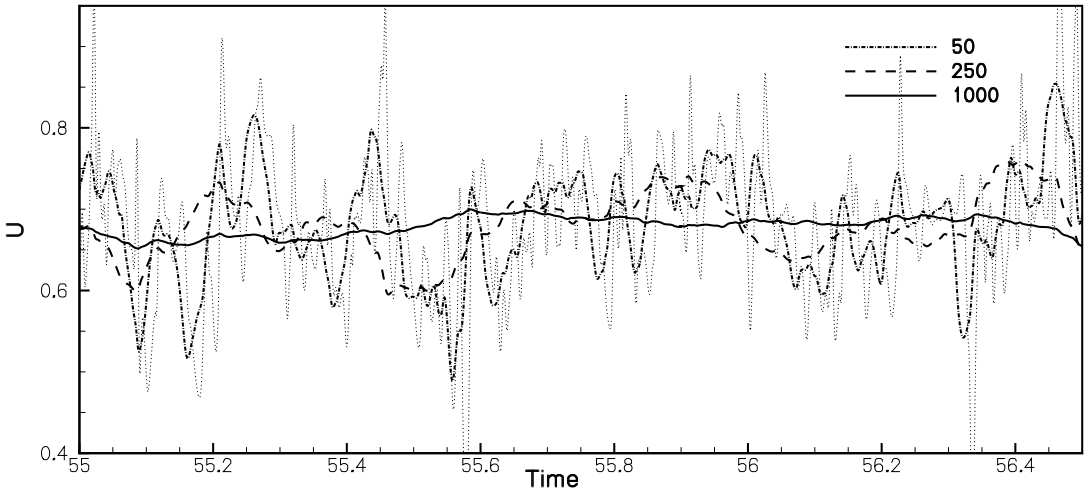
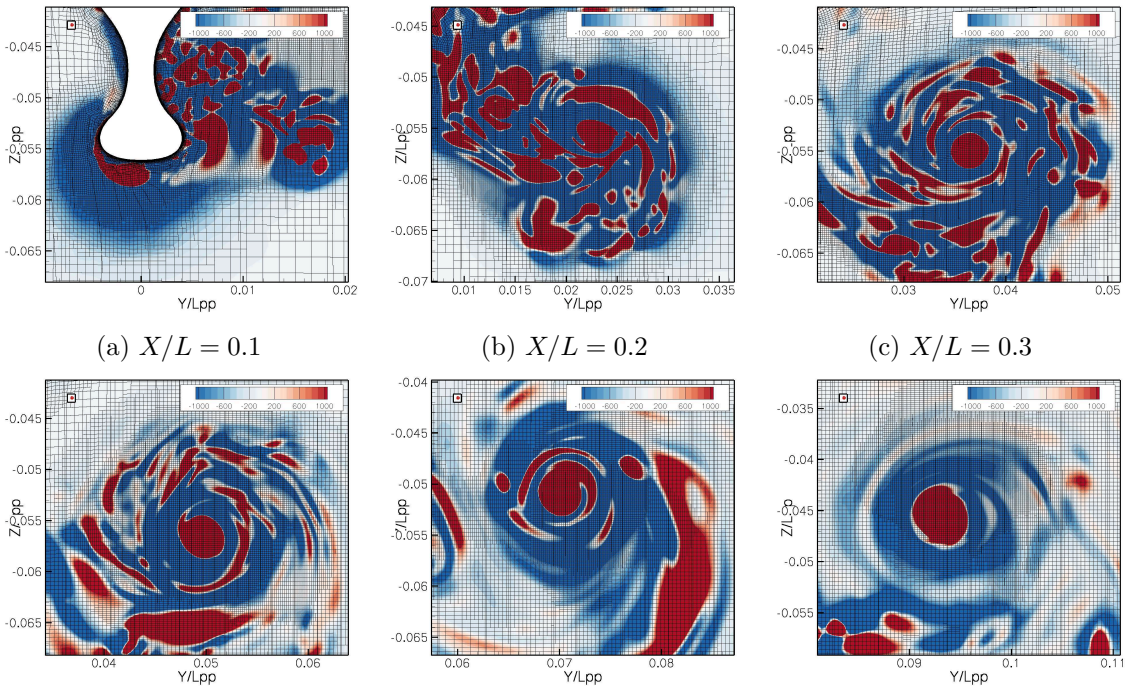


Figure 7.21 – Effect of different sliding averaging windows on the axial velocity in the SDTV core at $X/L = 0.12$

In order to switch from the interval averaging-based adaptation to the averaging over



(a) $X/L = 0.1$ (b) $X/L = 0.2$ (c) $X/L = 0.3$

(d) $X/L = 0.4$ (e) $X/L = 0.6$ (f) $X/L = 0.8$

Figure 7.22 – Views of the mesh and the instantaneous Q^* in cross-sections across the center of the SDTV, for DDES-SST. The small black box represents the estimated Taylor microscale, the red box the minimum cell size. The scale of all images is the same

the whole computational time, the simulation is stopped after 3800 time steps. According to what is discussed above, this computational time is almost sufficient for all upstream-created turbulence to pass along the hull and reach the back of the ship, while simultaneously the adapted mesh converges locally. The computation is then restarted using AGR based on averaging over the whole computational time, but as the jump of the number of cells in Figure 7.19b shows, AGR is not called right after the beginning of the computation. The first refinement is performed 0.6 non-dimensional units from the restart of the computation, to give the averaged criterion enough time to converge.

Finally, it is assessed whether the adapted mesh is sufficiently fine and specifically, if the goal of capturing all turbulent structures on at least four cells has been met. Figure 7.22 at the end of the simulation, confirms that the minimum cell size is maintained over the entire length of the SDTV and that the structures inside the vortex are well captured by the grid. Moreover, the outside ring vortices which are larger than those inside the vortex, are captured conveniently on the part of the grid that is refined less than the vortex core. For a region close to the sonar dome (Figure 7.22a) where some structures are smaller than 4 cells in thickness, a slightly finer mesh could have been beneficial. However, caution is needed for further refinement close to the sonar dome which may deteriorate the solution due to the effect of an unsuccessful WMLES.

7.7 Conclusion

In this chapter, the average-based adaptive grid refinement is used for the simulation of the DTMB 5512 in 10° static drift condition, with an EARSM RANS and a DDES-SST computation. The simulation conditions are similar to the study performed by Visonneau

et al. [104] for the same test case with a static mesh, which is used as a reference. The IIHR measurements of Yoon et al. [116] are also compared with the computed solutions.

To perform the adaptation, a combined free-surface and flux-component Hessian criterion is selected as the refinement criterion. The threshold for the Hessian criterion is set to $L/20$, based on experience for RANS simulation. The same value as for RANS is chosen for DDES. Based on the values used in [104], the minimum cell size for the refinement is set to capture all resolved eddies with at least four grid points. Finally, sliding window averaging with different interval sizes is applied for both simulations, while averaging over the entire computational time is used to stabilize the mesh at the end of the DDES computation. With these adaptation settings, all the major structures of the flow are captured by the mesh refinement. For DDES, the grid size is as small as the minimum cell size in the regions where the Hessian criteria are large, e.g. the vortex cores. In addition, the sliding average diminishes the high-frequency fluctuations yet leads to a reactive mesh which converges in little more than one non-dimensional flow time.

Both simulations, starting from an original grid of 15M cells, capture the global features of the flow like the main SDTV, BKTV and AFBV vortices by increasing the grid density around the regions where these features are located. In the EARSM simulation, the final adapted mesh has a lower number of cells (47M) than the reference simulation with fixed mesh (163M). Nevertheless, good distribution of fine cells leads to a solution which reproduces all global features of the experiments. In addition, in terms of the force coefficients, the obtained solutions with AGR have more or less the same differences with the experimental measurements as the reference simulation on a fixed mesh. However, the AGR EARSM solution is unsteady which it is formally not supposed to be. This unsteadiness which is observed around the sonar dome creates a meandering of the SDTV core which results in higher values for the parameters like TKE in the SDTV core. For the BKTV in contrary, the vortex and the shear layer from the bilge keel remain steady and therefore, as it is expected, the model minimizes the TKE in the vortex core.

In the DDES-SST simulation, due to the higher unsteadiness in the solution, the number of cells in the final adapted mesh increases to 154M. The cells around the vortices are much smaller and are refined to the minimum cell size in the vortex cores. In addition, for each given position in the DDES simulation, the adapted mesh converges locally almost as soon as the upstream resolved turbulence arrives. The much finer grids close to the hull are the reason for the high difference between the predicted force coefficients and the experimental data which is observed for the reference DDES simulation as well. These differences are probably the effect of an unsuccessful attempt of WMLES in regions close to the hull where the grids are fine. Both main vortices and their corresponding solutions in the vortex cores are as well represented on the adapted mesh as on the fixed grid. Moreover, the AGR maintains a high resolution of the local grids around the vortices up to the back of the ship contrary to the reference DDES solution where the fine refinement box covers the sonar dome wake until $X/L = 0.4$, and thus the accuracy of the predicted solution is preserved throughout the flow.

Under the influence of ring vortices in the SDTV, a core meandering occurs which creates much stronger velocity fluctuations in the vortex core than for EARSM. These fluctuations may be the reason for a maximum TKE in the core of the SDTV. However, in the core of the BKTV coming from the windward bilge keel, the TKE level is too low. Due to the presence of a steady shear layer on the front part of the bilge keel and the lack of sufficient instabilities to produce turbulence, it is likely that the flow suffers from modeled stress depletion. The TKE in the vortex core only rises when the shear layer becomes unsteady and as a result the resolved TKE increased. This confirms that the high TKE values in the vortex cores are an effect of unsteady flow features.

In summary, the average-based AGR DDES-SST simulation of static drift of the DTMB 5512 works well. Without the need for selecting a proper location for the refinement boxes which is the case for simulations on fixed-grids, the cell points are efficiently distributed to capture the local instabilities around the sonar dome, the SDTV, the bilge keels, the BKTV and the aft-body structures with the same number of cells as the fixed mesh with refinement boxes. In addition, due to the proper selection of the adaptation settings, the adapted mesh converges relatively fast to a mesh which is refined up to the desired minimum cell size around the vortex cores and all along the vortices.

Chapter 8

Conclusion and perspectives

The objective of this thesis is to facilitate the mesh generation process for hybrid RANS/LES simulations of realistic flows using adaptive grid refinement (AGR). However, the high impact of the local mesh resolution on the way that these models behave, creates issues when using the mesh adaptation. Refinement based on the instantaneous solution may switch the model behavior between RANS and LES, regardless of their suitability for the prediction of the physical phenomena which occur. In addition, the temporal and spatial local changes in the cell size can result in an incorrect representation of the resolved turbulence, which lead to physically wrong results in extreme cases.

The study of the mesh dependency for DES shows that for this turbulence model, using cells with smaller size does not necessarily lead to a more accurate solution. For an attached boundary layer, higher-resolution meshes extend the scope of operation of LES inside the boundary layer and reduce the modeled turbulence production. However, this change in the model behavior does not lead to the production of resolved turbulence and results in modeled stress depletion (MSD). The local mesh resolution also has an effect on the turbulence decay rate for a freely decaying turbulence test case. For smaller cells, higher dissipation of the modeled turbulence is observed and while the dissipation of the resolved turbulence is reduced, it does not change at the same rate. Moreover, a loss of resolved TKE occurs at the refinement interfaces. The fine turbulence structures coming from an upstream fine mesh cannot be resolved any longer on a coarse grid downstream. It is inevitable to have local refinement interfaces when AGR is used, especially for the hybrid RANS/LES solutions where, due to the local and extreme fluctuations in the velocity, many spatial and temporal changes are imposed on the mesh by the adaptation.

These mesh changes introduced by the adaptive grid refinement can be limited by adapting the mesh to a time-averaged solution. Averaging over an interval of the computed solution diminishes the effects of the intensive fluctuations. By selecting a proper averaging interval, high-frequency flow instabilities are represented in an average sense, while the main unsteady behavior of the flow is preserved in the time-evolution of the average solution. With a long averaging interval, most high-frequency fluctuations are averaged out and the averaged solution is changed slowly by the fluctuations, and in opposite, averaging over a shorter interval implies that the average solution evolves more rapidly due to these high-frequency fluctuations. Selecting an optimal averaging interval, on which the evolution of the average is based, is case dependent and the physics of the flow must be considered. For (quasi-)steady cases like backward-facing step flow, a low-frequency flow instability (e.g. the flapping motion) which has non-local effects on downstream will probably be conserved in the time evolution of the average. And for unsteady cases with body motion, like a ship in zigzag maneuver, the interval should be selected based on the frequency of the main macroscopic unsteady feature in the flow.

Tests concerning the limiting of the mesh changes by basing the AGR on the time-averaged solutions, reveal that the average-based adaptation is effective when the averaging process is applied over the instantaneous refinement criteria. Two strategies are considered for applying this averaging process: the entire-simulation averaging which averages over all the preceding instantaneous refinement criteria and the sliding window averaging which is applied only over the criteria of a specific number of previous time steps. Changing the length of the averaging interval affects the behavior of the AGR and leads to a different amount of refinement and mesh changes. The high-frequency fluctuations in the averaged solution as a result of a short averaging sliding window cause an increase in the mesh changes, particularly in the regions where these fluctuations are dominant. However, the adaptation responds quickly to changes in instantaneous solutions and a converged adapted mesh is obtained rapidly for (quasi-)steady flows. In opposite, for a longer averaging window, although more computing memory is needed to save the averaged solutions, the modification of the mesh is reduced and it converges gradually. Averaging over the whole computational time is a better option instead of a very long averaging interval, which increases significantly the computational costs, when the intent is to gradually average out all the fluctuations and obtain a stable mesh.

A first realistic test is successfully carried out on a DDES-SST simulation of a ship in drift. The simulations using the integrated average-based AGR in the ISIS-CFD solver reproduce the solutions of reference simulations on a static fine mesh. Although the adaptation parameters like the minimum cell size and the threshold are set based on the reference simulations, monitoring the variation of the number of cells during the first several refinements helps to ensure that these parameters are well chosen. The global behavior of the flow is well predicted on a converged adapted mesh with high grid density around the vortex cores, without the need to manually place refinement boxes. The maximum turbulence kinetic energy at the center of the vortices is consistent with the reference simulation and the experimental data.

In general, the adaptive refinement method presented is a good alternative to the manual construction of non-uniform meshes for complex geometries. The adaptation parameters like the minimum cell size and the threshold should be determined case by case and with caution. Furthermore, monitoring of the adaptation process is needed to avoid unnecessary mesh refinement. In addition, the averaging strategy can be chosen between an averaging sliding window and an averaging over the whole computational time, or even a combination of both. At the beginning, the adaptation based on the sliding window averaging method can be applied, which adapts the mesh actively according to the evolution of the flow. When a more or less converged mesh is achieved, the averaging strategy can be switched to an entire computation averaging to filter out the remaining slight mesh changes and to obtain a completely stable adapted mesh. The outcomes of this process is to reduce the influence of highly unsteady fluctuations on the mesh changes where AGR is used in hybrid RANS/LES simulations, to produce adapted meshes with a more or less static topology, which have the required quality for simulating real-life phenomena with various flow features.

Perspectives. It can be foreseen that the increasing attention to the simulation of realistic, complex flows using hybrid RANS/LES turbulence models will indicate the urgent need for an automatic adaptation process that improves and optimizes the local resolution of the mesh. The developed average-based AGR is a step towards achieving this. The adaptation approach could become an essential step of modeling any complex phenomena, which allows to obtain a converged adapted mesh instead of spending a lot of time and energy on the mesh generation process. However, like any newly developed model or

algorithm, the reliability of the average-based AGR must be proven so that it can be used without any doubt about the validity of its results. The average adaptation method has been tested only on one complex case so far. Therefore, it should be used in other realistic cases with different flow behavior and compared with references, to identify its strengths and weaknesses and prove its reliability.

As discussed above, since the minimum cell size and the threshold value are case-dependent, the determination of these parameters is the main difficulty of using AGR, especially if the characteristics of the flow are not well known. To choose the minimum cell size, the user can estimate the order of magnitude of the required smallest cell sizes according to the existing expressions for calculating different flow scales, knowing that for hybrid RANS/LES simulation, in general, the Taylor scale should be captured. However, for the threshold, developing a procedure which either determines automatically its value or gives an indication of its range would be helpful. For instance, automatic thresholding or an ability to change the thresholds during the computation could be a first step toward this goal. In addition, to increase the capabilities of the integrated AGR in ISIS-CFD, a feasibility study on the definition of adaptation for periodic boundary conditions can be carried out, in order to find a solution for the algorithmic difficulties in treating the periodic boundaries in parallel computation. The definition of this feature could eliminate the need for a larger computational domain in the spanwise dimension.

Further works can also be focused on improving the performance of the turbulence modeling. In DDES-SST simulations, effects of the modeled stress depletion and also the log-layer mismatch are detected in the solutions. IDDES simulations could be performed, which allow a WMLES computation in high-resolution regions close to the solid boundaries, to reduce the effect of the log-layer mismatch in the computed forces. In addition, regardless of the choice between the available DES-type models, these non-zonal models are influenced by the presence of the grey zones between the fully RANS and fully LES regions. The ongoing research subject of grey area mitigation (GAM) in order to trigger a rapid RANS-to-LES transition [67, 76, 88] could therefore be studied, to reduce the discontinuity of the turbulence production in the transition zone by injecting turbulent content at the RANS-LES interface. Grey area mitigation can narrow the transition zone and reduce the effect of MSD on the solution.

The scope of the developed adaptation procedure is not limited to DES-type hybrid RANS/LES simulations. For instance, in computations using the PANS turbulence models where the energy-based filtering decouples the closure model relation from the computational grid size, the averaged adaptation approach could still be used to obtain an optimal adapted mesh where the effects of mesh changes on the prediction of the resolved solution, as well as the perturbations of the solution, are kept low. In PANS, the cut-off filters are defined by the unresolved-to-total ratios of kinetic energy and dissipation [38] which determine the proportion of the resolved and the modeled parts of the solution. These two ratios are often selected globally and independently of the local cell sizes. Therefore, a mesh refinement study is always needed to confirm that the filter being used is suitable for the given grid size [10, 15], especially when these ratios are low and smaller cell sizes are required to resolve the small-scale structures. When the adaptation is employed, the mesh refinement study is possible by only changing the threshold [108]. However, in some cases, theses ratios are specified locally based on a relation between the local cell sizes and the local turbulent scales (e.g. [4, 65]). In these cases for complex geometries, the average-based AGR would lead to a dynamic modification of the model parameter ratios based on the adapted mesh. Although this approach leads to the dependency of the turbulence model on the local mesh resolution, it could provide the optimum modeling on the computational meshes that are employed.

Finally, in the framework of this thesis, the averaging approach was also found to be applicable in RANS computations. However, unsteadiness was observed in the RANS solution in regions where the flow is highly unstable and the RANS models are exposed to excessive refinement. Earlier studies on similar cases [110, 111] found that this was an effect of the turbulence model. To ensure that for the present case, this unsteadiness is only caused by the turbulence model and not the adaptation process, a simulation with another available RANS turbulence model like k-ω SST can be performed. If the effect is confirmed, then for similar unstable RANS simulations, the averaged refinement approach could be effective for mitigating the effect of the unsteadiness on the mesh.

In conclusion, to simulate complex phenomena in realistic conditions, it is necessary to use models and techniques to reduce computational costs while maintaining sufficient accuracy. The hybrid RANS/LES turbulence model and the averaged adaptation are a promising combination which makes the simulation of such phenomena affordable with today's computing technology.

Bibliography

[1] Abbas, N. and Kornev, N., "Computations of the Japan Bulk Carrier using URANS and URANS/LES methods implemented into OpenFOAM toolkit", in Tokyo Workshop on CFD in Ship Hydrodynamics, 2015.

[2] Alauzet, F. and Loseille, A., "High-order sonic boom modeling based on adaptive methods", *Journal of Computational Physics* 229:561–593, 2010.

[3] Babushka, I., Chandra, J., and Flaherty, J. E., "Adaptive methods for partial differential equations", vol. 39. Society for industrial & Applied Mathematics, Philadelphia, 1989.

[4] Basara, B., Krajnovic, S., and Girimaji, S., "PANS vs. LES for computations of the flow around a 3D bluff body", in 7^{th} International Symposium On Engineering Turbulence Modelling and Measurements, Lymassol, Cyprus, 2008.

[5] Bechara, W., Bailly, C., Lafon, P., and Candel, S. M., "Stochastic approach to noise modeling for free turbulent flows", *AIAA journal* 32:455–463, 1994.

[6] Belme, A., Alauzet, F., and Dervieux, A., "An a priori anisotropic goal-oriented error estimate for viscous compressible flow and application to mesh adaptation", *Journal of Computational Physics* 376:1051–1088, 2019.

[7] Belme, A., Dervieux, A., and Alauzet, F., "Time accurate anisotropic goal-oriented mesh adaptation for unsteady flows", *Journal of Computational Physics* 231:6323–6348, 2012.

[8] Ben-Nasr, O., Hadjadj, A., Chaudhuri, A., and Shadloo, M. S., "Assessment of subgrid-scale modeling for Large Eddy Simulation of a spatially-evolving compressible turbulent boundary layer", *Computers & Fluids* 151:144–158, 2017.

[9] Benard, P., Balarac, G., Moureau, V., Dobrzynski, C., Lartigue, G., and d'Angelo, Y., "Mesh adaptation for large-eddy simulations in complex geometries", *International Journal for Numerical Methods in Fluids* 81:719–740, 2016.

[10] Bensow, R. E. and Van Den Boogaard, M., "Using a PANS simulation approach for the transient flow around the Japan Bulk Carrier", *Journal of Ship Research* 63:123–129, 2019.

[11] Berger, M. J. and Colella, P., "Local adaptive mesh refinement for shock hydrodynamics", *Journal of Computational Physics* 82:64–84, 1989.

[12] Berger, M. J. and Oliger, J., "Adaptive mesh refinement for hyperbolic partial differential equations", *Journal of Computational Physics* 53:484–512, 1984.

[13] Borouchaki, H., George, P. L., Hecht, F., Laug, P., and Saltel, E., "Delaunay mesh generation governed by metric specifications. Part I. Algorithms", *Finite Elements in Analysis and Design* 25:61–83, 1997.

[14] Brandt, T. T., "Usability of explicit filtering in Large Eddy Simulation with a low-order numerical scheme and different subgrid-scale models", *International Journal for Numerical Methods in Fluids* 57:905–928, 2008.

[15] Busco, G. and Hassan, Y. A., "Solution verification of PANS model for a PWR fuel assembly", *Nuclear Engineering and Design* 353:110213, 2019.

[16] Castro-Díaz, M., Hecht, F., Mohammadi, B., and Pironneau, O., "Anisotropic unstructured mesh adaption for flow simulations", *International Journal for Numerical Methods in Fluids* 25:475–491, 1997.

[17] Chaouat, B. and Schiestel, R., "A new partially integrated transport model for subgrid-scale stresses and dissipation rate for turbulent developing flows", *Physics of Fluids* 17:065106, 2005.

[18] Chapelier, J.-B., De La Llave Plata, M., Renac, F., and Lamballais, E., "Evaluation of a high-order discontinuous Galerkin method for the DNS of turbulent flows", *Computers & Fluids* 95:210–226, 2014.

[19] Comte-Bellot, G. and Corrsin, S., "Simple Eulerian time correlation of full-and narrow-band velocity signals in grid-generated, isotropic turbulence", *Journal of Fluid Mechanics* 48:273–337, 1971.

[20] Davidson, L., "Using isotropic synthetic fluctuations as inlet boundary conditions for unsteady simulations", *Advances and Applications in Fluid Mechanics* 1:1–35, 2007.

[21] Daviller, G., Brebion, M., Xavier, P., Staffelbach, G., Müller, J.-D., and Poinsot, T., "A mesh adaptation strategy to predict pressure losses in LES of swirled flows", *Flow, Turbulence and Combustion* 99:93–118, 2017.

[22] De La Llave Plata, M., Couaillier, V., and Le Pape, M.-C., "On the use of a high-order discontinuous Galerkin method for DNS and LES of wall-bounded turbulence", *Computers & Fluids* 176:320–337, 2018.

[23] Deck, S., "Zonal-detached-eddy simulation of the flow around a high-lift configuration", *AIAA Journal* 43:2372–2384, 2005.

[24] Deck, S., "Recent improvements in the zonal detached eddy simulation (ZDES) formulation", *Theoretical and Computational Fluid Dynamics* 26:523–550, 2012.

[25] Deng, G. B., Leroyer, A., Guilmineau, E., Queutey, P., Visonneau, M., Wackers, J., and Del Toro Llorens, A., "Verification and validation of resistance and propulsion computation", in Tokyo Workshop on CFD in Ship Hydrodynamics, 2015.

[26] Deng, G. B. and Visonneau, M., "Comparison of explicit algebraic stress models and second-order turbulence closures for steady flow around ships", in 7^{th} Symposium on Numerical Ship Hydrodynamics, volume 10, page 5953, 1999.

[27] Ding, K., Fidkowski, K., and Roe, P. L., "Continuous adjoint based error estimation and r-refinement for the active-flux method", in 54^{th} AIAA Aerospace Sciences Meeting, page 0832, 2016.

[28] Duncan, D., "Special issue on grid adaptation in computational PDEs: Theory and Applications", *Applied Numerical Mathematics* 26, 1998.

[29] Durbin, P. A. and Reif, B. P., "Statistical theory and modeling for turbulent flows", 2^{nd} Ed. John Wiley & Sons, Ltd, United Kingdom, 2011.

[30] Durran, D. R., "The third-order Adams-Bashforth method: An attractive alternative to leapfrog time differencing", *Monthly Weather Review* 119:702–720, 1991.

[31] Duvigneau, R., Visonneau, M., and Deng, G. B., "On the role played by turbulence closures in hull shape optimization at model and full scale", *Journal of Marine Science and Technology* 8:11–25, 2003.

[32] Dwight, R. P., "Heuristic a posteriori estimation of error due to dissipation in finite volume schemes and application to mesh adaptation", *Journal of Computational Physics* 227:2845–2863, 2008.

[33] Eaton, J. K. and Johnston, J. P., "Low frequency unsteadiness of a reattaching turbulent shear layer", in Turbulent Shear Flows 3, pages 162–170, Springer, 1982.

[34] Flaherty, J. E., "Lecture notes in Finite Element Analysis", Rensselaer Polytechnic Institute, Dept. of Computer Science, 2000.
URL `http://www.cs.rpi.edu/ flaherje`

[35] Frey, P.-J. and Alauzet, F., "Anisotropic mesh adaptation for CFD computations", *Computer Methods in Applied Mechanics and Engineering* 194:5068–5082, 2005.

[36] George, P. L. and Borouchaki, H., "Delaunay triangulation and meshing - application to finite elements", Hermes, Paris, 1998.

[37] George, P. L., Hecht, F., and Vallet, M., "Creation of internal points in Voronoi's type method. Control adaptation", *Advances in Engineering Software and Workstations* 13:303–312, 1991.

[38] Girimaji, S. and Abdol-Hamid, K., "Partially-Averaged Navier Stokes model for turbulence: Implementation and validation", in 43^{rd} AIAA Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2005.

[39] Goodfriend, E., Chow, F. K., Vanella, M., and Balaras, E., "Large Eddy Simulation across a grid refinement interface using explicit filtering and reconstruction", in 20^{th} Symposium on Boundary Layers and Turbulence, Boston, Massachusetts, 2013.

[40] Goodfriend, E., Chow, F. K., Vanella, M., and Balaras, E., "Improving Large Eddy Simulation of neutral boundary layer flow across grid interfaces", *Monthly Weather Review* 143:3310–3326, 2015.

[41] Gritskevich, M. S., Garbaruk, A. V., and Menter, F. R., "Fine-tuning of DDES and IDDES formulations to the k-ω Shear Stress Transport model", *Progress in Flight Physics* 5:23–42, 2013.

[42] Gritskevich, M. S., Garbaruk, A. V., Schütze, J., and Menter, F. R., "Development of DDES and IDDES formulations for the k-ω Shear Stress Transport model", *Flow, Turbulence and Combustion* 88:431–449, 2012.

[43] Guilmineau, E., Deng, G. B., and Wackers, J., "Numerical simulation with a DES approach for automotive flows", *Journal of Fluids and Structures* 27:807–816, 2011.

[44] Hall, S., Behnia, M., Fletcher, C., and Morrison, G., "Investigation of the secondary corner vortex in a benchmark turbulent backward-facing step using cross-correlation particle imaging velocimetry", *Experiments in Fluids* 35:139–151, 2003.

[45] Hasan, M., "The flow over a backward-facing step under controlled perturbation: laminar separation", *Journal of Fluid Mechanics* 238:73–96, 1992.

[46] Hay, A., "Etude de stratégies d'estimation d'erreur numérique et d'adaptation locale de maillages non-structurés pour les équations de Navier-Stokes en moyenne de Reynolds", PhD thesis, Université de Nantes, 2004.

[47] Hay, A. and Visonneau, M., "Error estimation using the error transport equation for finite-volume methods and arbitrary meshes", *International Journal of Computational Fluid Dynamics* 20:463–479, 2006.

[48] Hu, R., Wang, L., and Fu, S., "Investigation of the coherent structures in flow behind a backward-facing step", *International Journal of Numerical Methods for Heat & Fluid Flow* 26:1050–1068, 2016.

[49] Hu, S., Zhang, C., Liu, H., Wang, F., and Li, Y., "IDDES simulation of flow separation on an 3D NACA23012 airfoil with spanwise ridge ice", in Atmospheric and Space Environments Conference, page 2862, Atlanta, Georgia, 2018.

[50] Jasak, H., "Error analysis and estimation for the finite volume method with applications to fluid flows", PhD thesis, Imperial College London, 1996.

[51] Kroll, N., Bieler, H., Deconinck, H., Couaillier, V., van der Ven, H., and Sorensen, K., "ADIGMA-A european initiative on the development of adaptive higher-order variational methods for aerospace applications", vol. 113. Springer Science & Business Media, Berlin, 2010.

[52] Kuate, R., "Anisotropic metrics for finite element meshes using a posteriori error estimates: Poisson and Stokes equations", *Engineering with Computers* 29:497–505, 2013.

[53] Leonard, A., "Energy cascade in Large-Eddy Simulations of turbulent fluid flows", *Adv. Geophys. A* 18:237–248, 1974.

[54] Leonard, B. P., "Simple high-accuracy resolution program for convective modelling of discontinuities", *International Journal for Numerical Methods in Fluids* 8:1291–1318, 1988.

[55] Li, X., Shephard, M. S., and Beall, M. W., "3d anisotropic mesh adaptation by mesh modification", *Computer Methods in Applied Mechanics and Engineering* 194:4915–4950, 2005.

[56] Loseille, A., Dervieux, A., and Alauzet, F., "Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations", *Journal of Computational Physics* 229:2866–2897, 2010.

[57] Loseille, A., Dervieux, A., and Alauzet, F., "Fully anisotropic goal-oriented mesh adaptation for 3D steady Euler equations", *Journal of Computational Physics* 229:2866–2897, 2010.

[58] Lund, T. S., Wu, X., and Squires, K. D., "Generation of turbulent inflow data for spatially-developing boundary layer simulations", *Journal of Computational Physics* 140:233–258, 1998.

[59] McRae, D. S., "r-refinement grid adaptation algorithms and issues", *Computer Methods in Applied Mechanics and Engineering* 189:1161–1182, 2000.

[60] Menter, F. R., "Zonal two equation κ-ω turbulence models for aerodynamic flows", in 23^{rd} Fluid Dynamics, Plasmadynamics, and Lasers Conference, page 2906, Orlando, Florida, 1993.

[61] Menter, F. R. and Egorov, Y., "The scale-adaptive simulation method for unsteady turbulent flow predictions. Part 1: theory and model description", *Flow, Turbulence and Combustion* 85:113–138, 2010.

[62] Menter, F. R. and Kuntz, M., "Adaptation of eddy-viscosity turbulence models to unsteady separated flow behind vehicles", in The aerodynamics of heavy vehicles: trucks, buses, and trains, pages 339–352, Springer, 2004.

[63] Menter, F. R., Kuntz, M., and Langtry, R., "Ten years of industrial experience with the SST turbulence model", *Turbulence, Heat and Mass Transfer* 4:625–632, 2003.

[64] Miller, T., Aref, P., Ghoreyshi, M., Jirasek, A., and Greenwood, R., "Adaptive mesh refinement for computing unsteady ship air wakes", in AIAA Aviation 2019 Forum, page 3031, 2019.

[65] Mirzaei, M., Krajnović, S., and Basara, B., "Partially-Averaged Navier–Stokes simulations of flows around two different Ahmed bodies", *Computers & Fluids* 117:273–286, 2015.

[66] Mitran, S. M., "A comparison of adaptive mesh refinement approaches for Large Eddy Simulation", Technical report, Washington Univ Seattle Dept. of applied mathematics, 2001.

[67] Mockett, C., Fuchs, M., Thiele, F., Wallin, S., Peng, S., Deck, S., Kok, J., van der Ven, H., Garbaruk, A., Shur, M. L., et al., "Non-zonal approaches for grey area mitigation", in Go4Hybrid: Grey Area Mitigation for Hybrid RANS-LES Methods, pages 17–50, Springer, 2018.

[68] Mozaffari, S., Visonneau, M., and Wackers, J., "Unsteady flow over a smooth flat plate using DES", in 20^{th} Numerical Towing Tank Symposium (NuTTS), Wageningen, The Netherlands, 2017.

[69] Mozaffari, S., Visonneau, M., and Wackers, J., "Average-based adaptive grid refinement in hybrid LES", in Direct and Large-Eddy Simulation XII, Madrid, Spain, 2019.

[70] Mozaffari, S., Visonneau, M., and Wackers, J., "Hybrid LES simulation of turbulent flows using adaptive grid refinement based on averaged quantities", in 8^{th} International Conference on Computational Methods in Marine Engineering (MARINE), Gothenburg, Sweden, 2019.

[71] Naddei, F., De La Llave Plata, M., Couaillier, V., and Coquel, F., "A comparison of refinement indicators for p-adaptive simulations of steady and unsteady flows using discontinuous Galerkin methods", *Journal of Computational Physics* 376:508–533, 2019.

[72] Nikitin, N., Nicoud, F., Wasistho, B., Squires, K., and Spalart, P. R., "An approach to wall modeling in large-eddy simulations", *Physics of Fluids* 12:1629–1632, 2000.

[73] Piomelli, U. and Balaras, E., "Wall-layer models for large-eddy simulations", *Annual Review of Fluid Mechanics* 34:349–374, 2002.

[74] Piomelli, U., Kang, S., Ham, F., and Iaccarino, G., "Effect of discontinuous filter width in large-eddy simulations of plane channel flow", *Studying turbulence using numerical databases XI* pages 151–162, 2006.

[75] Pons, K. and Ersoy, M., "Adaptive mesh refinement method. Part 1: Automatic thresholding based on a distribution function", 2019.
URL `https://hal.archives-ouvertes.fr/hal-01330679`

[76] Probst, A., Schwamborn, D., Garbaruk, A., Guseva, E., Shur, M. L., Strelets, M. K., and Travin, A. K., "Evaluation of grey area mitigation tools within zonal and non-zonal RANS-LES approaches in flows with pressure induced separation", *International Journal of Heat and Fluid Flow* 68:237–247, 2017.

[77] Przulj, V. and Basara, B., "Bounded convection schemes for unstructured grids", in 15^{th} AIAA Computational Fluid Dynamics Conference, page 2593, Anaheim, California, 2001.

[78] Queutey, P. and Visonneau, M., "An interface capturing method for free-surface hydrodynamic flows", *Computers & Fluids* 36:1481–1510, 2007.

[79] Rajasekaran, J., "On the flow characteristics behind a backward-facing step and the design of a new axisymmetric model for their study", Master's thesis, University of Toronto, 2011.

[80] Reuß, S., Knopp, T., Probst, A., and Orlt, M., "Assessment of local LES-resolution sensors for hybrid RANS/LES simulations", in Progress in Hybrid RANS-LES Modelling, pages 93–103, Springer, 2015.

[81] Rhie, C. and Chow, W. L., "Numerical study of the turbulent flow past an airfoil with trailing edge separation", *AIAA journal* 21:1525–1532, 1983.

[82] Rubino, G., "Large dynamic manoeuvers for a naval vehicle", Master's thesis, Ecole Centrale de Nantes, 2017.

[83] Sagaut, P., Deck, S., and Terracol, M., "Multiscale and multiresolution approaches in turbulence: LES, DES and hybrid RANS/LES methods: applications and guidelines", Imperial College Press, London, 2013.

[84] Schlüter, J., Pitsch, H., and Moin, P., "Large Eddy Simulation inflow conditions for coupling with Reynolds-averaged flow solvers", *AIAA journal* 42:478–484, 2004.

[85] Shur, M. L., Spalart, P. R., Strelets, M. K., and Travin, A. K., "Detached-eddy simulation of an airfoil at high angle of attack", in Engineering Turbulence Modelling and Experiments 4, pages 669–678, Elsevier, 1999.

[86] Shur, M. L., Spalart, P. R., Strelets, M. K., and Travin, A. K., "A hybrid RANS-LES approach with delayed-DES and wall-modelled LES capabilities", *International Journal of Heat and Fluid Flow* 29:1638–1649, 2008.

[87] Shur, M. L., Spalart, P. R., Strelets, M. K., and Travin, A. K., "Synthetic turbulence generators for RANS-LES interfaces in zonal simulations of aerodynamic and aeroacoustic problems", *Flow, turbulence and combustion* 93:63–92, 2014.

[88] Shur, M. L., Strelets, M. K., Travin, A. K., Probst, A., Probst, S., Schwamborn, D., Deck, S., Skillen, A., Holgate, J., and Revell, A., "Improved embedded approaches", in Go4Hybrid: Grey Area Mitigation for Hybrid RANS-LES Methods, pages 51–87, Springer, 2018.

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

[90] Spalart, P. R., "Comments on the feasibility of LES for wings, and on a hybrid RANS/LES approach", in 1^{st} AFOSR international conference on DNS/LES, Ruston, Louisiana, 1997.

[91] Spalart, P. R., "Detached-eddy simulation", *Annual Review of Fluid Mechanics* 41:181–202, 2009.

[92] Spalart, P. R. and Allmaras, S. R., "A one-equation turbulence model for aerodynamic flows", in 30^{th} Aerospace Sciences meeting and exhibit, Reno, Nevada, 1992.

[93] Spalart, P. R., Deck, S., Shur, M. L., Squires, K. D., Strelets, M. K., and Travin, A. K., "A new version of detached-eddy simulation, resistant to ambiguous grid densities", *Theoretical and Computational Fluid Dynamics* 20:181–195, 2006.

[94] Spazzini, P., Iuso, G., Onorato, M., Zurlo, N., and Di Cicca, G., "Unsteady behavior of back-facing step flow", *Experiments in Fluids* 30:551–561, 2001.

[95] Squires, K. D., Forsythe, J. R., and Spalart, P. R., "Detached-eddy simulation of the separated flow around a forebody cross-section", in Direct and Large-Eddy Simulation IV, pages 481–500, Springer, 2001.

[96] Strelets, M. K., "Detached eddy simulation of massively separated flows", in 39^{th} Aerospace Sciences Meeting and Exhibit, Reno, Nevada, 2001.

[97] Toosi, S. and Larsson, J., "Anisotropic grid-adaptation in Large Eddy Simulations", *Computers & Fluids* 156:146–161, 2017.

[98] Travin, A. K., Shur, M. L., Spalart, P. R., and Strelets, M. K., "Improvement of delayed detached-eddy simulation for LES with wall modelling", in European Conference on Computational Fluid Dynamics, ECCOMAS CFD, Egmond aan Zee, The Netherlands, 2006.

[99] Travin, A. K., Shur, M. L., Strelets, M. K., and Spalart, P. R., "Detached-eddy simulations past a circular cylinder", *Flow, Turbulence and Combustion* 63:293–313, 2000.

[100] Travin, A. K., Shur, M. L., Strelets, M. K., and Spalart, P. R., "Physical and numerical upgrades in the detached-eddy simulation of complex turbulent flows", in Advances in LES of complex flows, pages 239–254, Springer, 2002.

[101] Vanella, M., Piomelli, U., and Balaras, E., "Effect of grid discontinuities on large-eddy simulation statistics and flow fields", *Journal of Turbulence* page N32, 2008.

[102] Venditti, D. A. and Darmofal, D. L., "Anisotropic grid adaptation for functional outputs: application to two-dimensional viscous flows", *Journal of Computational Physics* 187:22–46, 2003.

[103] Visonneau, M., Deng, G. B., Guilmineau, E., Queutey, P., and Wackers, J., "Local and global assessment of the flow around the Japan Bulk Carrier with and without energy saving devices at model and full scale", in 31^{st} Symposium on Naval Hydrodynamics, pages 11–16, Monterey, California, 2016.

[104] Visonneau, M., Guilmineau, E., and Rubino, G., "Computational analysis of the flow around a surface combatant at 10 static drift and dynamic sway conditions", in 32^{nd} Symposium on Naval Hydrodynamics, Hamburg, Germany, 2018.

[105] Vogel, J. and Eaton, J. K., "Combined heat transfer and fluid dynamic measurements downstream of a backward-facing step", *Journal of Heat Transfer* 107:922–929, 1985.

[106] Wackers, J., Ait Said, K., Deng, G. B., Queutey, P., Visonneau, M., and Mizine, I., "Adaptive grid refinement applied to RANS ship flow computation", in 28^{th} Symposium on Naval Hydrodynamics, Pasadena, California, 2010.

[107] Wackers, J., Deng, G. B., Guilmineau, E., Leroyer, A., Queutey, P., and Visonneau, M., "Combined refinement criteria for anisotropic grid refinement in free-surface flow simulation", *Computers & Fluids* 92:209–222, 2014.

[108] Wackers, J., Deng, G. B., Guilmineau, E., Leroyer, A., Queutey, P., Visonneau, M., Palmieri, A., and Liverani, A., "Can adaptive grid refinement produce grid-independent solutions for incompressible flows?", *Journal of Computational Physics* 344:364–380, 2017.

[109] Wackers, J., Deng, G. B., Leroyer, A., Queutey, P., and Visonneau, M., "Adaptive grid refinement for hydrodynamic flows", *Computers & Fluids* 55:85–100, 2012.

[110] Wackers, J., Guilmineau, E., and Visonneau, M., "Unsteady behaviour in RANS simulation of the JBC and KVLCC2", in 20^{th} Numerical Towing Tank Symposium (NuTTS), Wageningen, The Netherlands, 2017.

[111] Wackers, J., Guilmineau, E., and Visonneau, M., "Using grid adaptation to understand ship flow instability", in International Conference on Adaptive Modeling and Simulation (ADMOS), Barcelona, Spain, 2017.

[112] Wackers, J., Jeanson, C. E., Queutey, P., Visonneau, M., Pellegrini, R., Serani, A., and Diez, M., "Hull shape optimisation using multi-fidelity metamodels and adaptive grid refinement", in 21^{st} Numerical Towing Tank Symposium (NuTTS), Cortona, Italy, 2018.

[113] Wang, J., Minelli, G., Dong, T., Chen, G., and Krajnović, S., "The effect of bogie fairings on the slipstream and wake flow of a high-speed train. An IDDES study", *Journal of Wind Engineering and Industrial Aerodynamics* 191:183–202, 2019.

[114] Wilcox, D., "Turbulence modeling for CFD", vol. 3. DCW Industries, La Cañada, California, 2006.

[115] Wissink, A., Kamkar, S., Pulliam, T., Sitaraman, J., and Sankaran, V., "Cartesian adaptive mesh refinement for rotorcraft wake resolution", in 28^{th} AIAA Applied Aerodynamics Conference, Chicago, Illinois, 2010.

[116] Yoon, H., Gui, L., Bhushan, S., and F., S., "Tomographic PIV measurements for a surface combatant at straight ahead and static drift conditions", in 30^{th} Symposium on Naval Hydrodynamics, Hobart, Tasmania, Australia, 2014.

Résumé

La modélisation des écoulements à des nombres de Reynolds élevés avec des phénomènes physiques complexes et réalistes constitue un défi en termes de ressources de calcul, malgré les développements importants en termes de méthodes de calcul et de puissance de calcul qui ont été réalisés ces dernières années. La grande variation de la taille des structures dans les écoulements turbulents réalistes implique, qu'une simulation complète de ces écoulements nécessite la résolution de toutes leurs échelles. L'erreur locale et la précision de ces simulations sont déterminées par la taille des cellules du maillage sur lequel les équations sont résolues. Pour obtenir des résultats satisfaisants permettant de prédire avec précision la physique complexe des problèmes, les cellules doivent être suffisamment petites. Le nombre requis de cellules pour le type de simulation où toutes les échelles sont résolues, appelé simulation numérique directe (Direct Numerical Simulation, DNS), dépasse de loin ce qui est abordable ou même réalisable par la technologie informatique actuelle. Par conséquent, l'application de ces simulations, coûteuses en temps de calcul, est limitée aux écoulements simples à faible nombre de Reynolds.

La modélisation de la turbulence est un moyen pour réduire les coûts de calcul en diminuant la complexité des solutions. L'idée est de simuler une partie de la solution en utilisant un modèle physique approximatif, au lieu de résoudre entièrement l'écoulement, afin d'obtenir un compromis entre les coûts de calcul et la précision des solutions. Des approches classiques pour modéliser la turbulence sont les modèles de Navier-Stokes en moyenne de Reynolds (RANS), abordables mais parfois peu fiables pour des écoulements complexes, et la LES (Large Eddy Simulation) qui est plus proche de la physique réelle mais qui, en contrepartie, est beaucoup plus coûteuse. En tant que compromis, les approches hybrides RANS/LES sont introduits comme modèles ayant des propriétés souhaitables pour les écoulements complexes. Ces modèles héritent des avantages des systèmes RANS et LES, afin de minimiser les ressources de calcul requises tout en préservant la précision de la solution. L'approche hybride RANS/LES est conçue pour se comporter comme un modèle RANS là où les structures tourbillonnaires à l'échelle dissipative dominent et où la solution RANS est fiable, comme par exemple un écoulement de couche limite attaché. Puisque les calculs avec cette méthode sont relativement peu coûteux, la simulation d'écoulements avec des géométries réalistes et des nombres de Reynolds élevés est possible. En revanche, lorsque RANS n'est pas assez précis, la partie LES du modèle est activée. Les écoulements avec des séparations sont un exemple de problème dans lequel RANS peut conduire à une solution physiquement incorrecte, tandis que l'approche LES prédit une solution plus précise, à condition que le maillage soit suffisamment fin localement. Pour des écoulements complexes réalistes dans lesquels existent les deux types de situation, un modèle de turbulence hybride RANS/LES est efficace pour réduire les coûts de calcul en agissant comme RANS, et en augmentant simultanément la précision en tant que modèle LES partout où ceci est nécessaire.

Pour tirer parti des modèles hybrides RANS/LES dans un problème réaliste, un maillage non-uniforme avec une résolution locale correcte partout dans le domaine, doit

être généré. Les propriétés locales du maillage doivent répondre aux exigences par rapport au comportement du modèle de turbulence, afin d'éviter une mauvaise utilisation de LES sur une grille insuffisamment fine, voir un maillage excessivement fin dans une région RANS. Par conséquent, pour obtenir un maillage optimal pour une simulation hybride RANS/LES, la construction itérative du maillage avec plusieurs simulations complètes, qui indiquent de mieux en mieux les tailles de cellules nécessaires localement afin de capturer les phénomènes physiques majeurs de l'écoulement, peut être nécessaire. Malheureusement, ceci nécessite beaucoup de temps et d'effort. Comme alternative, un processus de raffinement adaptatif de maillage (AGR) peut être utilisé pour adapter automatiquement le maillage en fonction des structures de l'écoulement. Par conséquence, une simulation peut commencer sur un maillage grossier et en raffinant ou déraffinant localement les cellules au cours de la simulation, un maillage optimal avec une résolution élevée uniquement dans les régions où ceci est requise, sera obtenu. Cette technique réduit les efforts de l'utilisateur pour la génération de maillages, ce qui représente un gain de temps important surtout pour les écoulements complexes.

Cependant, contrairement aux simulations RANS dans lesquelles le raffinement adaptatif de maillage est utilisé de manière fiable depuis plusieurs années, pour un modèle de turbulence hybride RANS/LES, toute modification des propriétés du maillage affecte les performances du modèle de turbulence. Dans cette thèse de doctorat, la combinaison de la technique d'adaptation de maillage avec des simulations hybrides RANS/LES est évaluée, afin de développer une approche de raffinement adaptatif qui convient pour ces modèles de turbulence. L'étude de la dépendance par rapport au maillage pour le modèle de turbulence DES (Detached Eddy Simulation) montre que, pour ce modèle de turbulence, l'utilisation de cellules de plus en plus petites ne conduit pas nécessairement à une solution plus précise. Pour une couche limite attachée, les maillages de très haute résolution élargissent le domaine d'application du modèle LES à l'intérieur de la couche limite et réduisent par conséquent la production de turbulence modélisée. Toutefois, ce changement de comportement du modèle ne conduit pas à la production de turbulence résolue, ce qui entraîne une diminution de la tension modélisée (Modeled Stress Depletion, MSD). Les tailles locales des cellules ont également un effet sur le changement du taux de décroissance de la turbulence, pour un cas-test de turbulence en dissipation libre. Dans les cellules les plus petites, on observe une dissipation plus élevée de la turbulence modélisée, ce qui n'est compensé que partiellement par la décroissance ralentie de la turbulence résolue. De plus, une perte d'energie cinétique turbulente résolue se produit aux interfaces entre petites et grandes cellules. Les petites structures turbulentes en provenance d'un maillage fin en amont, ne peuvent plus être résolues sur un maillage plus grossier en aval. Et puisqu'il est inévitable d'avoir des interfaces de raffinement locales lors de l'utilisation de l'AGR, en particulier pour les solutions hybrides RANS/LES où, en raison des fluctuations locales et extrêmes de la vitesse, de nombreux changements spatiaux et temporels sont imposés au maillage par l'adaptation, il est certain que ce type de problème apparaisse dans une simulation réaliste.

Le principal défi de cette thèse consiste à trouver une méthode de raffinement qui ne soit pas affectée par cet effet perturbateur des modifications du maillage. En raison de la forte instationnarité des solutions hybrides RANS/LES, le raffinement basé uniquement sur la solution instantanée entraîne un changement significatif du maillage à chaque adaptation et intensifie la perturbation que le changement de maillage impose à la solution. Par conséquent, le choix est fait de baser l'adaptation sur la solution moyennée dans le temps, afin de créer des maillages adaptés ayant une topologie plus ou moins statique avec le moins de modifications possible. Par conséquent, une étude est faite des effets du moyennage en temps sur la solution moyennée. Cette étude révèle qu'en sélectionnant un intervalle

de moyennage approprié, le comportement instationnaire principal de l'écoulement est incluse dans l'évolution temporelle de la solution moyenne, tandis que les instabilités à haute fréquence de l'écoulement sont représentées sous forme moyennée. La sélection d'un intervalle de moyennage optimal, sur lequel sera basé l'évolution de la moyenne, dépend du cas et pour ce choix, la physique de l'écoulement doit être prise en compte. Pour les cas (quasi-) stationnaires, comme l'écoulement autour d'une marche déscendante, une instabilité de basse fréquence (comme par exemple, le mouvement de battement) qui a des effets non-locaux sur l'écoulement en aval, doit probablement être pris en compte dans l'évolution temporelle de la solution moyenne. Et pour les cas instables comme un navire en zigzag ou une aile à angle d'attaque variable lors d'une manoeuvre, l'intervalle doit être choisi en fonction de la fréquence des évolutions instationnaires macroscopiques dans l'écoulement.

Différentes approches de raffinement basées sur plusieurs stratégies de moyennage en temps sont testées pour obtenir une procédure d'adaptation optimale. L'adaptation basée sur la moyenne est efficace lorsque le processus de calcul de la moyenne est appliqué sur les critères de raffinement instantanés calculés à chaque étape temporelle. Au contraire, le calcul d'un critère de raffinement basé sur l'écoulement moyen ne permet pas d'obtenir une précision suffisante et ne semble pas pertinent pour capturer l'écoulement instationnaire à chaque instant. Deux stratégies sont envisagées pour appliquer ce processus de calcul de la moyenne: la moyenne sur la simulation complète, ce qui implique de moyenner tous les critères de raffinement instantanés précédents, et la moyenne dans une fenêtre glissante qui est appliquée uniquement aux critères calculés lors d'un nombre spécifique de pas de temps précédents. Changer la longueur de l'intervalle de moyennage affecte le comportement de l'AGR et conduit à un changement du taux de raffinement du maillage. Les fluctuations à haute fréquence de la solution moyennée à cause d'une courte fenêtre glissante se traduisent par une augmentation des modifications du maillage, en particulier dans les régions où ces fluctuations sont dominantes. Cependant, l'adaptation réagit plus rapidement aux solutions instantanées et un maillage adapté convergé est obtenu plus rapidement pour les écoulements (quasi-) stationnaires. Au contraire, pour une fenêtre de moyennage plus longue, plus de mémoire est nécessaire afin de stocker l'ensemble des solutions à moyenner, mais la modification du maillage est réduite et elle converge progressivement. Lorsque l'objectif est d'enlever progressivement toutes les fluctuations et d'obtenir un maillage stable, la moyenne sur l'ensemble du temps de calcul est une meilleure option qu'un très long intervalle glissante de moyennage, car cette dernière augmente considérablement les coûts de calcul.

L'adaptation basée sur des critères de raffinement moyennés est intégrée dans le solveur fluide ISIS-CFD dont la discrétisation est basée sur des volumes finis non structurés. La méthode est testée avec succès sur une simulation DDES-SST d'un navire en dérapage. Les simulations utilisant l'AGR intégré basé sur la moyenne reproduisent les solutions de simulations de référence sur un maillage fin statique. Bien que les paramètres d'adaptation, tels que la taille de cellule minimale et le seuil, soient définis en fonction des simulations de référence, le suivi de la variation du nombre de cellules au cours des premiers raffinements permet de s'assurer que ces paramètres sont bien choisis. Le comportement global de l'écoulement est bien prédit sur un maillage adapté fin avec une densité de grille élevée autour des centres des tourbillons, sans qu'il soit nécessaire de placer manuellement des zones de raffinement. Des simulations RANS et DDES sont possibles avec le même choix de paramètres pour le raffinement automatique, l'évolution du critère de raffinement venant de la dynamique accrue de l'écoulement produit par la DDES est suffisant pour produire le maillage qui est nécessaire pour résoudre l'écoulement turbulent instationnare. En outre, il est démontré que ce maillage est suffisamment fin pour représenter la dynamique des petites

structures et permettre une analyse physique de leur comportement, en comparant par exemple l'évolution de deux tourbillons avec des origines différentes. Le niveau d'énergie cinétique turbulente maximale au centre des tourbillons correspond à la simulation de référence et aux données expérimentales.

En général, la méthode de raffinement adaptatif présentée ici constitue une bonne alternative à la construction manuelle de maillages non-uniformes pour des géométries complexes. La stratégie de calcul de la moyenne peut être sélectionnée entre une fenêtre glissante et une évaluation basée sur la totalité du temps de calcul, voire même une combinaison des deux. Au début d'une simulation, l'adaptation basée sur le calcul dans une fenêtre glissante peut être appliquée, ce qui permet d'adapter de façon dynamique le maillage, en fonction de l'évolution de l'écoulement. Quand un maillage plus ou moins convergé est atteint, la stratégie de calcul de la moyenne peut être basculée vers l'évaluation basée sur la durée entière de la simulation, pour écarter les petits changements de maillage restants et, par conséquent, obtenir un maillage adapté parfaitement stable. Le résultat de ce processus est une réduction de l'influence des fluctuations instationnaires qui apparaissent dans les solutions hybrides RANS/LES sur l'évolution du maillage due au raffinement automatique. Cette combinaison prometteuse d'adaptation de maillage et modélisation hybride de la turbulence permet de simuler les phénomènes complexes des écoulements réalistes à un prix abordable avec la technologie informatique d'aujourd'hui.

CENTRALE NANTES

Titre : Raffinement de maillage adaptatif pour les modèles de turbulence hybrides RANS/LES

Mots clés : RANS/LES hybride; Raffinement de maillage adaptatif; Critère moyen.

Résumé : Les modèles hybrides RANS/LES, qui utilisent les capacités des modèles RANS et LES, conviennent pour la simulation d'écoulements à nombre de Reynolds élevé avec des phénomènes physiques et des géométries complexes. Pourtant, en raison de la dépendance du comportement de ces modèles par rapport à la taille locale des mailles, la génération de maillages avec une bonne résolution est cruciale. Pour une simulation hybride RANS/LES complexe, la génération du maillage est une étape qui demande un effort important. Un processus de raffinement adaptatif est une alternative intéressante, mais nécessite la prise en compte des effets du changement de maillage sur la performance du modèle.
Cette thèse porte sur le développement du raffinement adaptatif de maillage basé sur du moyennage et son intégration dans le solveur fluide ISIS-CFD. Le but est d'obtenir un maillage adapté ayant une topologie globalement statique, qui suit les caractéristiques principales de l'écoulement.
Premièrement, la dépendance des modèles hybrides RANS/LES par rapport aux modifications du maillage et l'effet d'une interface de raffinement sur la production et la destruction de turbulence sont étudiés. Ensuite, un moyennage en temps des solutions instantanées basé sur différents intervalles est effectué pour filtrer les fluctuations dans la solution moyenne, en fonction des fréquences caractéristiques de l'écoulement. Une procédure de raffinement est ensuite développée, basée sur deux stratégies de moyennage des critères de raffinement instantanés: le moyennage dans une fenêtre glissante et le moyennage sur la totalité du temps de calcul. Le processus d'adaptation proposé est évalué en effectuant une simulation DDES d'un navire en dérive, afin de reproduire la solution sur un maillage fin de référence. L'adaptation basée sur la moyenne réussit à reproduire les principales caractéristiques de l'écoulement et crée des maillages raffinés stables autour des tourbillons principaux. Ainsi, l'approche adaptative constitue une alternative au processus manuel de génération de maillage pour les simulations hybrides RANS/LES, en particulier avec des géométries complexes.

Title : Adaptive Grid Refinement for Hybrid RANS/LES

Keywords : Hybrid RANS/LES; Adaptive grid refinement; Averaged criterion.

Abstract : Taking advantage of the capabilities of RANS and LES models, hybrid RANS/LES models are suitable for the simulation of high Reynolds number flow with complex physical phenomena and geometries. However, due to the dependency of the behavior of these models on the local grid size, the generation of mesh with the right resolution is crucial. For a complex hybrid RANS/LES simulation, the mesh generation is a time- and effort-consuming step. An adaptive refinement process is an attractive alternative, but requires the consideration of mesh change effects on the performance of the model.
This thesis focuses on the development of adaptive grid refinement based on averaging and its integration in the ISIS-CFD flow solver. The aim is to obtain an adapted mesh which has a generally static topology based on the main flow features.
First, the dependency of hybrid RANS/LES models on the mesh changes and the effect of a refinement interface on the turbulence production and destruction are considered. Then, time-averaging of the instantaneous solutions over various intervals is carried out to filter the flow fluctuations in the mean solution based on the frequencies of the flow features. A refinement procedure is then developed based on two averaging strategies over instantaneous refinement criteria: the sliding window averaging, and the averaging over the whole computational time. The proposed adaptation process is assessed by performing a DDES-based simulation of a ship in drift in order to reproduce the solution on a reference fine mesh. The average-based adaptation successfully follows the main flow features and creates stable refined grids around the main vortices. Thus, the adaptive approach can be an alternative for the manual mesh generation process of hybrid RANS/LES simulations, especially with complex geometries.