Mathematical and Numerical Modeling of Turbulent Flows

JOÃO M. VEDOVOTO[1], **RICARDO SERFATY**[2] and **ARISTEU DA SILVEIRA NETO**[1]

[1]Faculdade de Engenharia Mecânica, Universidade Federal de Uberlândia,
Avenida João Naves de Ávila, 2121, Sta. Mônica, 38400-902 Uberlândia, MG, Brasil
[2]CENPES-PETROBRAS, Cidade Universitária, Avenida 1, Quadra 7, sala 2118,
Ilha do Fundão, 21941-915 Rio de Janeiro, RJ, Brasil

ABSTRACT

The present work is devoted to the development and implementation of a computational framework to perform numerical simulations of low Mach number turbulent flows over complex geometries. The algorithm under consideration is based on a classical predictor-corrector time integration scheme that employs a projection method for the momentum equations. The domain decomposition strategy is adopted for distributed computing, displaying very satisfactory levels of speed-up and efficiency. The Immersed Boundary Methodology is used to characterize the presence of a complex geometry. Such method demands two separate grids: An Eulerian, where the transport equations are solved with a Finite Volume, second order discretization and a Lagrangian domain, represented by a non-structured shell grid representing the immersed geometry. The in-house code developed was fully verified by the Method of Manufactured Solutions, in both Eulerian and Lagrangian domains. The capabilities of the resulting computational framework are illustrated on four distinct cases: a turbulent jet, the Poiseuille flow, as a matter of validation of the implemented Immersed Boundary methodology, the flow over a sphere covering a wide range of Reynolds numbers, and finally, with the intention of demonstrating the applicability of Large Eddy Simulations - LES - in an industrial problem, the turbulent flow inside an industrial fan.

Key words: Turbulent flows, Large Eddy Simulation, Immersed Boundary Method, Low-Mach Number Flows, Industrial Flows.

INTRODUCTION

Fluid dynamics is present in an enormous amount of natural phenomena as well as problems of physics, biology, engineering, among other areas. There are two ways to analyze such problems, through material and virtual experimentation. The first one requires material modeling, with the construction of benches and models of the physical problems, as well as instrumentation for gathering information. The second requires mathematical modeling, which leads to models with differential equations, integral equations and / or integro-differential equations. The virtual experimentation, provides valuable information about the problem. Moreover, their solution has become a very powerful way of understanding the turbulent characteristic of fluid flow, its phenomenology and consequences. The methods involved in virtual experimentation of fluid

Correspondence to: João Marcelo Vedovoto
E-mail: jmvedovoto@mecanica.ufu.br

flow requires computational modeling, which consists in transforming the mathematical models that can generate numeric algorithms and quantitative information about the problem at hand. Once a mass of results is generated, one should treat the calculated data in order to perform the visualization of the simulations results and statistical analysis of the numerical experiment. It is clear, therefore, that the procedure adopted in virtual experiment is similar to that of materials testing. Both forms of experimentation are seen today with equal potential analysis. Both should still be seen as complementary to one another, and should be widely used together, offering a huge potential for analysis and understanding of natural phenomena. The virtual experiments, when applied to problems involving fluid motion, give rise to what is called Computational Fluid Dynamics, or CFD.

The physical nature of the engineering problems of interest is characterized as being multidisciplinary and multi-scale. As aforementioned, the analysis every problem involving fluid dynamics have a common step, which is the modeling of fluid movement. By adding additional phenomena, as for instance fluid structure interactions, additional models, e.g. the structural dynamics modeling, are required. Another class, are reactive flows which fall into the combustion problems. In this case, it is required the further modeling of chemical reactions, involving the addition of hundreds of differential equations in addition to those related to the movement of the fluid itself. The physical principles used for the development of mathematical modeling are presented in the literature (White 1999) as: conservation of mass, second Newton's law and conservation of energy. Based on these principles and the continuum hypothesis, the balance of mass, linear momentum and energy are give rise to the so-called equations of continuity, Navier-Stokes (for Newtonian fluids) and energy respectively. Additional equations may be required when additional models are used, for example for the analysis of structural behavior or in the presence of chemical reactions. What should result are models that are composed by the number of equations and unknowns, such as velocity, pressure, temperature, density, concentration, among others.

Turbulent flows are omnipresent in academic and industrial applications, and, although such a phenomena may be part of our everyday life, turbulence is an area of knowledge in which research is always needed. It may be said that the main contribution of the present work is to provide a mathematical and numerical framework suitable for the simulation of complex, turbulent flows over a wide range of industrial applications, with advantage of retain a Cartesian grid for the solution of the Navier-Stokes equations.

The solution of the models arising from the mathematical modeling require methodologies that fit to the characteristics of the problem analyzed and must be representative of these problems. The present work relies on the formulation for compressible, low Mach number flows. The solution of this type of model demands a pressure-velocity coupling. A projection method based on the fractional step methodology is used. The discretization of the differential model leads to the appearance of linear systems, which are solved in the present work by resorting to iterative solvers like the Msip (Schneider and Zedan 1981) among others. In summary, the present work is presented by the following sequence: an introduction, a literature review, mathematical modeling, computational modeling, verification of computer code, validation and presentation of results of virtual experimentation of fluid iteration structure problems. It is worth noting that the present work is the product of a fruitful scientific collaboration between the Fluid Mechanics Laboratory (MFLab) of the Federal University of Uberlândia and Petrobras.

MATHEMATICAL MODELING

In this section the mathematical framework developed for simulating turbulent flows is presented. The modeling involves the numerical solution of partial differential equations where the LES

methodology was chosen to model turbulence. We will show, in the present section, the balance equations that model the flows under consideration, as well as the hypothesis adopted to proceed with their closures. Since the LES methodology is retained, the filtering process of the balance equations is also described in details.

DIFFERENTIAL MATHEMATICAL MODEL

The main objective of the present work is to model and simulate turbulent low Mach number flows. Considering a mathematical model suitable for variable density flows, in which the primary transported variables are the density $\rho(\vec{x}, t)$, the three velocity components $u_i(x, t)$ $(i = 1, 2, 3)$ and the temperature $T(\vec{x}, t)$. The balance equations for these variables in space $(x_i, i = 1, 2, 3)$ and time (t) are summarized below, along with an equation of state that relates the thermodynamic pressure to the density and temperature.

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0 \, , \tag{1}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j} \tau_{ij} + f_i \, , \tag{2}$$

$$\frac{\partial \rho c_p T}{\partial t} + \frac{\partial \rho u_j c_p T}{\partial x_j} = -\frac{\partial}{\partial x_j} \left(\lambda \frac{\partial T}{\partial x_j} \right) + \dot{Q} + S_T \, , \tag{3}$$

$$P = P_o(t) + P'(\mathbf{x}, t); \quad P_o(t) = \rho R T \, ; \tag{4}$$

in the above equations, τ_{ij} stand for the viscous tensor, f_i represent any body forces, S_T is the source term of temperature. The quantities c_p and λ stand for the constant-pressure specific heat and the thermal conductivity respectively. $\dot{Q}$ is source term of internal energy generation. $P_o(t)$ is the thermodynamic pressure, a function of time only in the flows of interest. $P'(\mathbf{x}, t)$ is the fluid-static pressure, henceforth denoted only by p. Since the flows treated in this work operate in open domains, the thermodynamic pressure does not vary in time, hence, $P_o(t)$ becomes a reference pressure, e.g. atmospheric pressure. $R = R^o \Sigma_{k=1}^{K} Y_k / \mathcal{M}_k$ is the gas constant, with R^o, and T is the being the universal gas constant. The molar mass of species are denoted by $\mathcal{M}_k$ temperature. Considering a Newtonian fluid the viscous tensor is given by,

$$\tau_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu \frac{\partial u_k}{\partial x_k} \delta_{ij} \, , \tag{5}$$

where μ is the fluid's dynamic viscosity and δ_{ij} is the Kronecker delta tensor.

The decomposition of the pressure shown in (Eq. 4) is reinforced here. In the work of Lessani and Papalexandris (2006) it is explained that there are basically three considerations to be addressed when dealing with the time derivative of pressure: (i) the system is open, (ii) the system is closed and (iii) the system is semi-open so that there is a restricted opening (for example, a crack) between the flow domain and its exterior. Concerning the energy dissipation due to the action of viscous forces, with the assumption of low Mach number flow, this term also vanishes. Based on the assumptions and hypothesis listed above, the system of equations to be solved can finally be written,

$$\frac{\partial \rho}{\partial t} + \frac{\partial \rho u_i}{\partial x_i} = 0 \, , \tag{6}$$

$$\frac{\partial \rho u_i}{\partial t} + \frac{\partial \rho u_i u_j}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \tau_{ij}}{\partial x_j} \tag{7}$$

$$\frac{\partial \rho c_p T}{\partial t} + \frac{\partial \rho u_j c_p T}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\lambda \frac{\partial T}{\partial x_j}\right) , \tag{8}$$

$$\rho = \rho(T) \tag{9}$$

LARGE EDDY SIMULATION

As mentioned in the introduction of this work, the solution of the balance equations (mass, momentum and energy, in the temperature form) is deemed sufficient to represent the evolution of a multi-component gas-phase mixture in either laminar or turbulent flow, provided that the continuum hypothesis holds, and once suitable constitutive equations for the fluids of interest are provided (Haworth 2010). The full solution of this set of equations, however, is far from being straightforward, specially under turbulent regime. To solve each and every scale of a turbulent flow, i.e., to perform a Direct Numerical Simulation of a turbulent flow, with the nowadays computational power available is out of reach for problems of interest in engineering, see for instance Pope (2000). Since all length and timescales of turbulence must be resolved, the computational cost of a DNS increases proportionally to $Re^{9/4}$, where Re is the Reynolds number.

The highly non-linear set of Partial Differential Equations PDE's are extremely sensitive to small variations of initial and boundary conditions. If the simulation of an actual device is of interest, the boundary condition information are seldom available with the required precision.

Moreover, the mesh required for DNS simulation of turbulent flows must be fine enough to resolve the smallest structures of the flow.

To give an idea of the computational cost of a DNS, Poinsot and Veynante (2005) present the results of a DNS of a premixed flame front at atmospheric pressure interacting with isotropic turbulence. The mesh adopted in such simulation contains approximately two million grid points and the corresponding computational domain is a cubic box of size $5 \times 5 \times 5$ mm^3. Finally, even if a computation of full set of PDE's of a turbulent reactive flow is computationally possible, the large amount of data generated is too cumbersome to deal with, for most practical purposes.

Directly opposed to the completeness of Direct Numerical Simulations description, the Unsteady Reynolds Averaged Navier Stokes (URANS) computations provide a kind of extreme filtering process of the flows properties of interest. The cut-off length scale of this filtering process is of order of the flow's integral scale. Such a process results in almost averaged equations, with unclosed terms, where some sort of closure is needed. These unclosed terms are comprised by the Reynolds stress tensor and the turbulent scalar fluxes. Although indisputably much less expensive than DNS, the fact of providing only mean values prevents an URANS model of predicting turbulent structures that may influence a fluid structure interaction, for instance.

Large Eddy Simulations (LES) can be thought as an intermediate methodology between DNS and URANS. The large scales of turbulent flows are explicitly calculated, whereas the non-linear interactions between large scales and subgrid scales are modeled using closure models. The balance equations for large eddy simulations are obtained by filtering the instantaneous transient equations (6-8). LES can determine the instantaneous nature of a turbulent "large scale" but a subgrid model is required to account for the effects of small turbulent scales over the physical processes (Poinsot and Veynante 2005). Since large structures are explicitly computed in LES, instantaneous turbulence characteristics are quite well captured.

In comparison with a URANS methodology, LES may provide a better prediction of the turbulence physical characteristics, as well as better statistical results.

Filtered transport equations

The main idea behind the LES methodology is to compute the largest structures of the flow field, typically larger then the computational grid, whereas the non-linear interaction between large and subgrid scales must be modeled. This methodology has been widely used with success for several kinds of flows, see in Pope (2000), Lesieur and Métais (1996), Germano et al. (1991), Lilly (1992), Silveira-Neto et al. (1993).

In LES, the dependent variables can be filtered in spectral space, where components greater than a given cut-off frequency are suppressed (Da-Silva 2001), or in physical space (weighted filtering over a given small volume). In the present work a box filter in physical space is adopted. These types of LES filters have been discussed, among others, in Pope (2000). A filtered quantity f is defined as,

$$\bar{f}(x) = \int f(x')F(x - x')dx',$$ (10)

where F is the LES filter function, defined as:

$$F(x) = \begin{cases} 1/\Delta^3 & if \, |x_i| \leq \Delta, \, i = 1, 2, 3 \\ 0 & otherwise \end{cases},$$ (11)

where (x_1, x_2, x_3) are the spatial coordinates of the vector x, and $\Delta = \sqrt[3]{V_c}$ is a cubic box of size equivalent to that of the finite-volume grid used for discretizing the transport equations in the Eulerian field. V_c is the volume of a mesh element. It is worthy noting that the filter used in the present work is implicit, i.e., the process of filtering is implicitly performed based on the fact that the actual process of discretization of the transport equations is, intrinsically, a filtering process in space and time. Thus, the characteristic length of the box in (Eq. 11) is equivalent to the characteristic length of the finite volume grid adopted in the solution of the discretized system of equations and the temporal band width is associated with the size of the time step used in the simulations.

In variable density flows it is useful to define the mass-weighted Favre filtering:

$$\bar{\rho}\widetilde{f}(x) = \int \rho f(x')F(x - x')dx'.$$ (12)

The filtered quantities $\bar{f}$ and/or $\widetilde{f}$ are computed by the numerical simulations. The fluctuations $f' = f - \bar{f}$ are the unresolved, subgrid scale part of the quantity f. With such a definition it is possible to perform the filtering over the transport equations. This procedure, however, must be done with care, once, contrary to RANS averaging for instance, in LES $\bar{f}' \neq 0$. Filtered and double filtered functions are not equal neither $\bar{\bar{f}} \neq \bar{f}$.

The filtered transport equations can be written,

$$\frac{\partial \rho}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_i}{\partial x_i} = 0 \,,$$ (13)

$$\frac{\partial \bar{\rho}\tilde{u}_i}{\partial t} + \frac{\partial \bar{\rho}\tilde{u}_i\tilde{u}_j}{\partial x_j} = -\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial \bar{\tau}_{ij}}{\partial x_j} - \frac{\partial \bar{\tau}_{ij}^{SCG}}{\partial x_j} \,,$$ (14)

$$\frac{\partial \overline{\rho c_p}\widetilde{T}}{\partial t} + \frac{\partial \overline{\rho c_p}\tilde{u}_i\widetilde{T}}{\partial x_j} = -\frac{\partial}{\partial x_j}\left(\bar{\lambda} \frac{\partial \widetilde{T}}{\partial x_j} \right) - \frac{\partial Q_{Tj}}{\partial x_j} + \bar{S}(T),$$ (15)

where, $\bar{\tau}_{ij}^{SCG} = (\overline{\rho \widetilde{u_i u_j}} - \bar{\rho} \tilde{u}_i \tilde{u}_j)$ and $Q_{Ti} = (\overline{\rho \widetilde{u_i T}} - \bar{\rho} \tilde{u}_i \tilde{T})$ are the subgrid scale (SGS) stress tensor and the SGS scalar fluxes, respectively. The SGS closure is associated with unknown components $\bar{\tau}_{ij}^{SCG}$ and Q_{Ti}. Therefore, the system of equations (13 - 15) is open and need to be closed.

In the next section the modeling approaches adopted in the present work for the terms $\bar{\tau}_{ij}^{SCG}$ and Q_{Ti} are described.

Turbulence closure using the Smagorinsky model

Most of the subgrid models assume that the smallest turbulent scales are approximately isotropic and homogeneous, exhibiting a "universal" character, whereas the largest, more energized structures, are dependent on the flow characteristics. The universal character of the smallest scales should render them more amenable to modeling, i.e., such an universal behavior of turbulent scales should make the subgrid model more universal. The most popular subgrid scale model is the Smagorinsky model (Smagorinsky 1963). The unresolved momentum flux are expressed according to the Boussinesq assumption (Ferziger and Peric 1996, Germano et al. 1991, Poinsot and Veynante 2005),

$$\bar{\tau}_{ij}^{SCG} - \frac{\delta_{ij}}{3} \bar{\tau}_{ij}^{SCG} = \nu_{SGS} \left(\frac{\partial \tilde{u}_i}{\partial x_j} + \frac{\partial \tilde{u}_j}{\partial x_i} - \frac{2}{3} \frac{\partial \tilde{u}_k}{\partial x_k} \delta_{ij} \right) = 2\nu_{SGS} \left(\tilde{S}_{ij} - \frac{\delta_{ij}}{3} \tilde{S}_{kk} \right), \tag{16}$$

where ν_{SGS} is the subgrid (or turbulent) viscosity and $\tilde{S}_{ij}$ is the strain rate tensor of the resolved field (Ferziger and Peric 1996).

In the Smagorinsky model the eddy viscosity ν_{SGS} is obtained by assuming that the small scales are in equilibrium, so that energy production and dissipation are in balance. This yields,

$$\nu_{SGS} = 2\bar{\rho}(C_s \Delta)^2 |\tilde{S}| = 2\bar{\rho}(C_s \Delta)^2 \left(2\tilde{S}_{ij} \tilde{S}_{ij} \right)^{1/2}, \tag{17}$$

where C_s is known as the Smagorinsky constant. Its value is chosen, in such a way that local equilibrium is maintained, between 0:1 to 0:25 (Germano et al. 1991, Pope 2000). Although its successful application in numerous cases, the Smagorinsky model is restricted, with some limitations. Among then, it can be presented: (i) the model does not prevent a null turbulent viscosity in cases when re-laminarization of the flow occurs, an important effect that may occur in reactive flows, once the thermal expansion can locally lead to flow re-laminarization; (ii) by assuming the equilibrium hypothesis, the model fails in regions where the flow is in transition to turbulence; (iii) the model does not allow the energy transfer from the smallest to the largest structures, which is known as backscattering (Lilly 1992); (iv) for simulations where grid elements are highly anisotropic, this model tends to be excessively diffusive, and the accuracy of the numerical methods may be compromised (Scotti et al. 1993); (v) it is known that the Smagorinsky model can be excessively diffusive, especially near walls. One possible solution to circumvent this last drawback is to diminish the value of C_s near walls, thus limiting the nearwall eddy viscosity. In this respect Ferziger and Peric (1996) present the van Driest damping function,

$$C_s = C_{so} \left(1 - e^{-n^+/A^+} \right)^2, \tag{18}$$

where n^+ is the normal distance from the wall in viscous wall units, $n^+ = nu_\tau / \nu$, and the shear velocity $u_\tau = \sqrt{\tau_w / \rho}$ and τ_w is the shear stress at the wall. A^+ is a constant, usually taken as 25. Ferziger and Peric (1996) argue that although this modification of C_s produces the desired result, it is difficulty to justify its use within the LES context. The SGS model should only depend on the local properties of the flow, and a distance to a wall is not a flow property.

Turbulence closure using the dynamic Smagorinsky Model

Large eddy simulation has been developed and used as a prediction tool for turbulent flows during the past forty years. This method experienced advances with the development of the dynamic Smagorinsky model (Germano et al. 1991), with the modifications proposed by (Lilly 1992). Based also on the Boussinesq hypothesis, the dynamic procedure uses coefficients that are automatically computed using information contained in the resolved turbulent scales, thereby eliminating possible uncertainties associated with tunable parameters. One great advantage of the dynamic model is that in the modeling of flows with presence of walls, the turbulent viscosity tends to zero automatically towards the wall, once the Leonard tensor L_{ij}, which depends only on the velocity field, tends to zero at the walls. This fact makes the use of damping functions, like the van Driest damping function showed above, no longer necessary.

Several authors have been using the dynamical Smagorinsky model in a wide range of applications, see for instance Lesieur et al. (2005), Meneveau et al. (2006), Moin (2008). Moin et al. (1991) applied the dynamic procedure to scalar transport and subgrid kinetic energy models for compressible turbulent flows using Favre filtering.

Similarly to the Smagorinsky model, the subgrid viscosity is given by,

$$\mu_{SGS} = \bar{\rho} C_s \Delta^2 |\widetilde{S}| = \bar{\rho} C_s \Delta^2 \left(2\widetilde{S}_{ij}\widetilde{S}_{ij} \right)^{1/2}. \tag{19}$$

In the dynamic Smagorinsky model the function $C_s(\vec{x}, t)$ is evaluated using a dynamical procedure. Adopting the nomenclature used in the work of Pierce (2001), the following notation for density-weighted test filtering is used:

$$\widehat{\widetilde{u}} = \frac{\widehat{\overline{\rho}\widetilde{u}}}{\widehat{\bar{\rho}}}. \tag{20}$$

In dynamic modeling, the test filter width is denoted by $\widehat{\Delta}$ and is usually taken to be twice the width of the grid filter, Δ. For the subgrid turbulent stress model, Eq. (16), the dynamic procedure gives,

$$C_s = \frac{L_{ij}M_{ij}}{2M_{ij}M_{ij}}, \tag{21}$$

where,

$$\mathcal{L}_{ij} = \left(\widehat{\overline{\rho}\widetilde{u}_i\widetilde{u}_j} - \bar{\rho}\widehat{\widetilde{u}}_i\widehat{\widetilde{u}}_j \right), \tag{22}$$

and

$$\mathcal{M}_{ij} = \widehat{\bar{\rho}}\widehat{\Delta}^2\widehat{\widetilde{S}}_{ij}|\widehat{\widetilde{S}}| - \overline{\rho}\widehat{\Delta^2 2\widetilde{S}_{ij}|\widetilde{S}|}. \tag{23}$$

It is noteworthy that subgrid models are generally valid for predicting statistical properties of the subgrid scales but usually cannot account for instantaneous subgrid scale fluctuations (Pierce 2001). For the dynamic procedure to be applicable, the quantity to be modeled must vary substantially between the grid and test filter scales; otherwise, the difference between the values of a property in the test filter and in the grid filter will not be significant and therefore may not be used for modeling subgrid-scale quantities. Examples of quantities that cannot be modeled dynamically are dissipation and chemical reaction rates, because these phenomena occur almost exclusively at the smallest scales, which are always unresolved in LES.

NUMERICAL MODELING

In this section, the numerical procedures chosen to solve the transport equations presented above are described. The solution of an equation originally proposed under the continuum hypotheses in a discrete domain, invariably leads to a loss of information. However, it is possible to minimize such loss, by choosing appropriately the numerical methods to solve the equations. Important factors that directly impact the performance of a numerical method that seeks to solve the Navier-Stokes equations are: (*i*) the choice of the variables arrangement in the computational grid, (*ii*) the rate of convergence and the accuracy of the numerical scheme chosen to perform the discretization in space of both viscous and advective terms, and (*iii*) the rate of convergence and the accuracy of the method chosen to perform the temporal integration.

In general lines, the numerical method chosen for solving the variable density momentum, temperature and Poisson equations for pressure correction is based on a three-dimensional, conservative, staggered, finite-volume discretization. The central difference scheme (CDS) is applied to express the diffusive contributions of the transport equations. For the advective contributions both CDS and Deferred Correction (Ferziger and Peric 1996) approaches were implemented and are available in the developed numerical code. A fully implicit approach is adopted, and the resulting linear systems are solved using the MSIP - Modified Strongly Implicit Procedure (Schneider and Zedan 1981) for velocity components. Since the developed numerical code adopts a pressure-velocity coupling based on pressure variations, an appropriate algorithm for pressure-velocity coupling is needed. Here, a projection method based on the fractional steps technique is used, resulting in a variable-coefficient Poisson equation which is solved with the Bi-Conjugate Gradient Stabilized solver, BICGSTAB (van der Vorst 1981).

It is worthy to emphasize that although the mathematical modeling presented is suitable for compressible flows, where normally there is no need for a pressure-velocity coupling, i.e., the thermodynamic pressure is naturally and strongly coupled with the velocity field, we are interested in low Mach number flows, where the density variations may arise from strong temperature variations and therefore it may not be neglected (Ferziger and Peric 1996).

Once the mathematical models to be used are defined, we must define the numerical framework retained for the simulation of the Low-Mach number flows we are interested. The next section reports the essential features of the numerical solver FLUIDS 3D that has been used to conduct the numerical simulations reported hereafter. The reader may refer to Vedovoto et al. (2011) for further insights on the Eulerian solver.

The present numerical simulations are conducted with a central difference scheme (CDS) to represent the spatial derivatives. Time integration is performed using the backward difference scheme (BDF) with a CFL number value set to 0:5. Further information about the available discretization procedures, as well as the verification of the numerical code developed can be found in Vedovoto et al. (2011).

MATHEMATICAL MODELING OF THE IMMERSED BOUNDARY METHOD
FOR LOW-MACH NUMBER VARIABLE DENSITY FLOWS

In this section the mathematical and numerical models retained for the implementation of the immersed boundary methodology, using the multi-direct forcing method is reviewed. Focus is given now for its modeling concerning variable-density low Mach number flows and in the use of such a methodology in distributed computing. The verification and validation of the methodology are presented.

In the present work a fully implicit approach is used for the transport equations, i.e., all the transport equations solved on the Eulerian domain are discretized according to the temporal integration described in Vedovoto et al. (2011). For instance, re-arranging the different terms, the momentum equation, once discretized in time, takes the following form:

$$\frac{\alpha_2(\rho u_i)^{n+1} + \alpha_1(\rho u_i)^n + \alpha_0(\rho u_i)^{n-1}}{\Delta t} = -\frac{\partial_p^{n+1}}{\partial_{x_i}}$$
$$+ (\theta_{2,\,n+1})m_2 + (\theta_{1,\,n+1})m_1 + (\theta_{0,\,n+1})m_0 \tag{24}$$

where $\theta_{0,\,n+1}$, $\theta_{1,\,n+1}$ and $\theta_{2,\,n+1}$ are parameters for the time integration technique chosen, Vedovoto et al. (2011), and,

$$m_{k+1} = \left(\frac{\partial \tau_{ij}^{n+k}}{\partial x_j} - \frac{\partial \rho^{n+k} u_i^{n+k} u_j^{n+k}}{\partial x_j} \right), \; k = -1, 0, 1 \tag{25}$$

It is worthy re-calling that Eq. (7) and consequently Eq. (24), are filtered transport equations, ready to be used in a Large Eddy Simulation. The variables are shown without the filtering notation ($\bar{}$) or ($\tilde{}$) only for the sake of conciseness. In order to demonstrate the mathematical and numerical models used in the Immersed Boundary and the Direct and Multi-direct Forcing methods, Eq. (24) is rewritten as:

$$\frac{\alpha_{2,\,n+1}(\rho u_i)^{n+1} + \alpha_{1,\,n+1}(\rho u_i)^{n+1} + \alpha_{0,\,n+1}(\rho u_i)^{n-1}}{\Delta t} = \frac{\partial p^{\,n+1}}{\partial x_i}$$
$$+ (\theta_{2,\,n+1})m_2 + (\theta_{1,\,n+1})m_1 + (\theta_{0,\,n+1})m_0 + f_i = RHS + f_i, \tag{26}$$

If a temporary term is added and subtracted in the temporal term of Eq. (26) this equation can be rewritten as:

$$\frac{\alpha_{2,\,n+1}(\rho u_i)^{n+1} + \alpha_{2,\,n+1}\rho u_i^* - \alpha_{2,\,n+1}\rho u_i^* + \alpha_{1,\,n+1}(\rho u_i)^n + \alpha_{0,\,n+1}(\rho u_i)^{n-1}}{\Delta t} = RHS + f_i, \tag{27}$$

Applying an algorithm similar to the one retained in the fractional step method (Vedovoto et al. 2011), the Eq. (27) can be decomposed in two equations:

$$\frac{\alpha_{2,\,n+1}\rho u_i^* + \alpha_{1,\,n+1}(\rho u_i)^n + \alpha_{0,\,n+1}(\rho u_i)^{n-1}}{\Delta t} = RHS, \tag{28}$$

$$\frac{\alpha_{2,\,n+1}(\rho u_i)^{n+1} - \alpha_{2,\,n+1}\rho u_i^*)}{\Delta t} = f_i. \tag{29}$$

Through the Immersed Boundary Methodology, the solid-fluid interface forces are evaluated by enforcing momentum to the fluid particles over the interface fluid-solid. Such a methodology is particularly suitable for problems involving fluid-structure interactions, once the difficulty of re-meshing the computational grid is circumvented by the use of two independent domains. The transport equations of fluid flow are solved in an Eulerian domain (fixed, Cartesian for instance), while the immersed geometries are represented by a set of Lagrangian points.

The Lagrangian force field is evaluated by the Direct Forcing Method, which was proposed by Wang et al. (2008). One of the characteristics of such method is that ad-hoc constants are not necessary, therefore the modeling of non-slip condition on immersed interface is physically consistent and general, i.e. the Direct

Forcing Method can be applied to a wide range of fluid flow problems, without the need of adjustments. By the continuum hypothesis, at any point k, over the surface of an immersed object the momentum at such a point can be evaluated by means of Eq. (29). However, it is useful to rewrite Eq. (29) for points only over the immersed boundary:

$$\frac{\alpha_{2,\,n+1}\left(\rho U_i^{n+1} - \rho_k U_i^*\right)}{\Delta t} = F_i. \tag{30}$$

In Eq. (30) the capital letters stand for variables evaluated at each Lagrangian point.

Although for the flows of interest the continuum hypothesis is valid, variables like ρu_i of Eq. (29) and $\rho_k U_{i,k}$ of Eq. (30) have not the same value, because in a discretized finite-volume computational grid, they can be in different positions even if they are hold in the same control volume. As aforementioned, the coupling between information that comes from Eulerian and Lagrangian domains is the main feature of the Immersed Boundary methodology. Moreover, the coupling between both domains, through the correct evaluation of the terms f_i and $F_{i,k}$, is essential for the correct prediction of flow characteristics. Such a coupling is based on discrete versions of a Dirac delta function.

The source term, f_i, defined over the whole Eulerian domain (henceforth denoted by Ω), is null, excepting in the regions where the control volumes coincide with the immersed geometry, enabling the Eulerian field to perceive the presence of a solid interface. Eq. (31) displays such a behavior.

$$f_i = \begin{cases} F_i & if \ x_i = X_i \\ 0 & if \ x_i \neq X_i \end{cases} \tag{31}$$

where x_i is the position of a fluid particle and X_i is the position of a fluid particle placed over a Lagrangian point at the solid interface. It is worthy to note that through Eq. 31 the field f_i is a discontinuous function, but mathematically defined over all the Ω domain. If a staggered framework of discretization is used for the fluid equations, such a coincidence never happens once the primary variables of the fluid are positioned in different locations. When the flow of interest have complex geometries within the computational domain (Ω) it is necessary to distribute the function F_i on its neighborhoods. This is achieved by replacing the definition given by eq. (31) by a discrete interpolation/distribution function. There are several forms of such a function. A detailed study of their form and efficiency can be found in Griffith and Peskin (2005).

PROCEDURES OF INTERPOLATION OF VELOCITIES AND DISTRIBUTION OF DISCRETE FIELD FORCES

The set of equations Eq. (28) and Eq. (29) have to be solved separately to take into account all viscous and body forces in the flow field. Firstly Eq. (28) is solved by the numerical methods already described in Vedovoto et al. (2011), briefly aforementioned. The numerical methods used for solving such system of equations are not in the scope of the present work and can be found in details in Vedovoto (2011). Here we will focus on how Eq. (29) is solved and the numerical aspects associated to the algorithms used.

Since Eqs. (29) and (30) provide equivalent results and the Lagrangian force necessary to modify the fluid flow must be applied at each Lagrangian point k, it is natural to evaluate such forces in the Lagrangian domain, i.e, by solving Eq. (30). In order to solve this equation there are two important factors that must be taken into account: the first one is the term ρU_i^{n+1}, which stand for the given velocity at the immersed boundary. It can be clearly noted that such a velocity is other-else known or prescribed, e.g, if the immersed object is not moving, the velocity of a fluid particle at the interface must be zero, therefore, in that case, $U_i^{n+1} = 0$. If the boundary is moving with a given velocity U_w, where U_w is the given velocity at wall, $U_i^{n+1} = U_w$.

The other term in Eq. (30) is the term ρU_i^*. This is, a priori, not known since the Lagrangian point over the interface may not lie over the exact location where the velocity u_i^* is held. The solution for such a drawback is to interpolate the velocity values near a Lagrangian point k using a weight function:

$$U_i^* = \sum_\Omega u_i^* D_h \Delta V, \tag{32}$$

where ΔV is the volume of control volume where the Lagrangian point k lies within. D_h if a weight-pondered function, which has the form:

$$D_h = \prod \frac{\varphi \left[(X - x_i)/\Delta_{x_i} \right]}{\Delta_{x_i}}; \tag{33}$$

and,

$$\varphi = \begin{cases} |1 - r| & if \, r \leq 1, \\ 0 & if \, r > 1. \end{cases} \tag{34}$$

r is the normalized distance between a Lagrangian point and the center of an Eulerian control volume where such Lagrangian point lies.

Several functions can be used as distance-based interpolation, assuming different shapes like Gaussian, cubic etc., and different support sizes, (Griffith and Peskin 2005). The function retained in the present work is a hat function (Eqs. 33 and 34), which gives a second order rate of convergence, with the advantage of requiring only one computational cell around the Lagrangian point k.
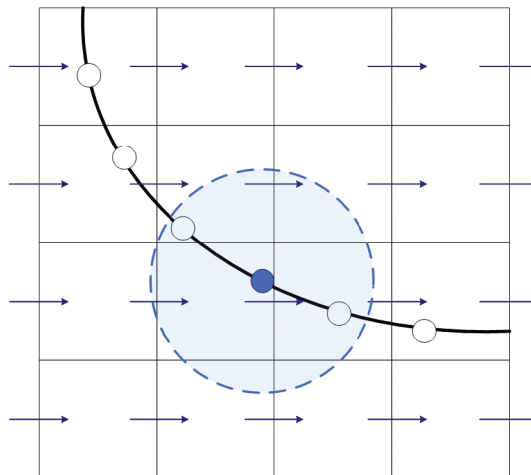
Figure 1 - Immersed geometry on a staggered finite volume computational grid. Interpolation of the Eulerian velocity field to a Lagrangian point.

Figure (1) shows a two-dimensional scheme of an immersed boundary on a staggered finitevolume computational grid. It can be noted that the Lagrangian point - represented by the dots - are not coincident with the faces of the control volumes. The interpolation of the Eulerian velocities in this case is needed. Using the hat function as the interpolation function, velocity values that lies within the blue circle shown in figure (1) will be interpolated to the Lagrangian point at the center of the circle. In a procedure similar to the interpolation of the "Eulerian" velocities to the Lagrangian points, once Eq. (30) is evaluated, it is necessary to distribute the Lagrangian forces F_i to the Eulerian control volumes. This is made with aid of Eq. (35).

$$f_i = \sum_k F_i \, D_h \, \Delta A_k * \Delta_{Sk} \tag{35}$$

In three-dimensional simulations the immersed object is represented by a superficial mesh composed by triangles, each one with a characteristic area, ΔA_k, and characteristic length ΔS_k, (Vedovoto 2007). Note that the interpolation procedure interpolates variables that can be positioned at faces, like velocities components, or at cell centers, like density for instance. Once the velocities are interpolated and the Lagrangian force is evaluated, such a force can also be distributed to faces or cell centers. It depends on the arguments used in Eqs. (33) and (34). In the present work, the Lagrangian forces are distributed to the cell faces.
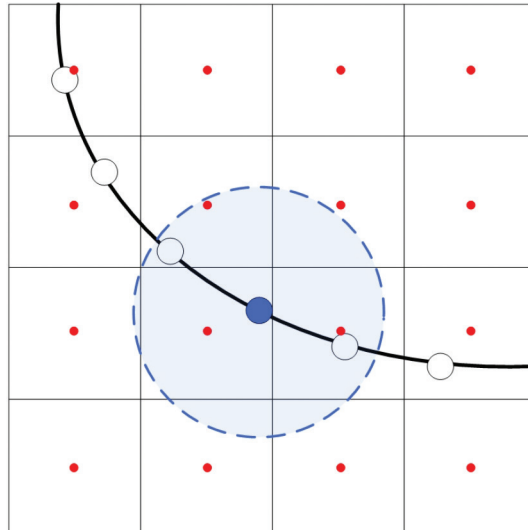
Figure 2 - Immersed geometry on a staggered finite volume computational grid. Distribution of the Lagrangian forces to the Eulerian field.

In Fig (2), it is displayed the area (doted circle) in which the Lagrangian force evaluated at a point k is distributed over a Eulerian grid. It is noteworthy that a same Eulerian cell can receive contributions of more than one Lagrangian point, being these contributions summed.

Concerning the parallelization of the immersed boundary method, two kinds of flows must be considered. The flows under fluid-structure interaction, in which the immersed boundary moves within the computational domain, and therefore may cross sub-domains boundaries, and flows where a sub-domain boundary can cross an immersed object, but such a geometry is still. For the latter, the algorithm is quite simple. It is enough that in the beginning of the simulation each processor reads a file where the immersed geometry is described, and then, it is verified which Lagrangian points lies within each sub-domain. Evidently, it may give a different number of Lagrangian points per processor, and static load balancing must be carefully evaluated.

In the interpolation/distribution procedures, that are the base of the immersed boundary method used in the present work, it is important to keep in mind that, in distributed computing, all the primitive variables must be continuous, and it is valid for the interpolated Lagrangian velocities U, and for the Eulerian distributed forces f_i as well, i.e, it is important to communicate correctly such information across sub-domains in order to keep the overlapping cells always with updated values. Considering that communication is required in each cycle of the multidirect forcing, such a communication must be as efficient as possible.

If we consider figure (3), it is possible to see that information must be interpolated from and distributed to overlapping cells. Since primitive variables, like velocity components, must be interpolated to a Lagrangian point k, from an overlapping cell, it causes no further difficulties, once at the moment of the interpolation, the velocity field is already updated in the overlapping cells, and therefore, any primitive variable can be interpolate to a Lagrangian point k regardless the subdivisions of the computational domain (Vedovoto 2011).
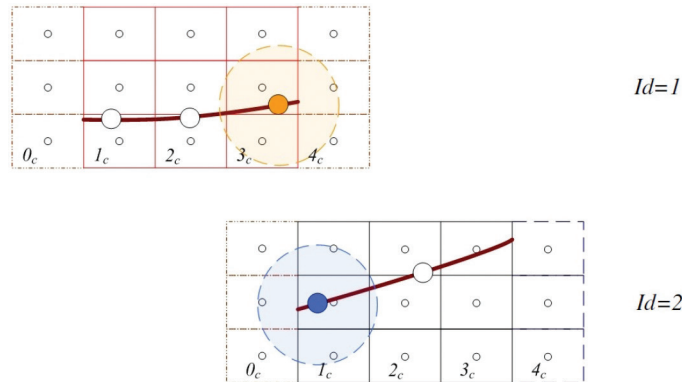
Figure 3 - Correct distribution of Lagrangian forces in a parallel approach. Step 1.

The procedure for distributing the Lagrangian force from the point *k* to a region of the Eulerian grid is a little more tricky however. It can be noted in figure (3), that a portion of the Lagrangian force once concentrated in a point *k* is distributed in overlapping cells in both processors IDs 1 and 2. One must remember that distributed Eulerian force is the result of a sum of the contributions of different Lagrangian points that lies within a given region (see Eq. 35). Once stated that it is possible that the contribution of a computational cell that lies close to an sub-domain edge must receive the contribution of each Lagrangian point that have a $f_i \neq 0$ from its neighboring sub-domain, an operation of communication and summation of the force f_i must be performed, as shown in figure (4). The result of such an operation is that the values of forces that were once known in column 1*c* of *Id* = 2, is now known in the overlapping cell of *Id* = 1.
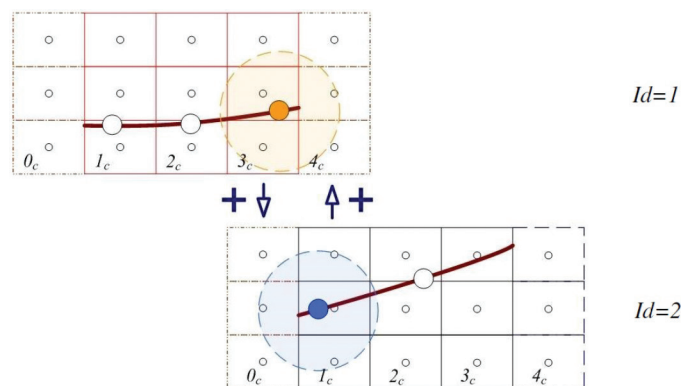
Figure 4 - Correct distribution of Lagrangian forces in a parallel approach. Step 2.

As a final part of the correct distribution of forces among different sub-domains, it is necessary now to sum the contribution of the force distributed in the overlapping cell of $Id = 1$ for example in the cell $1c$ of $Id = 2$. Once the value of force in the overlapping cell of $Id = 1$ is already the correct value of Lagrangian force at that point, it is enough the communicate back the information to $Id = 2$, as shown in figure (5).
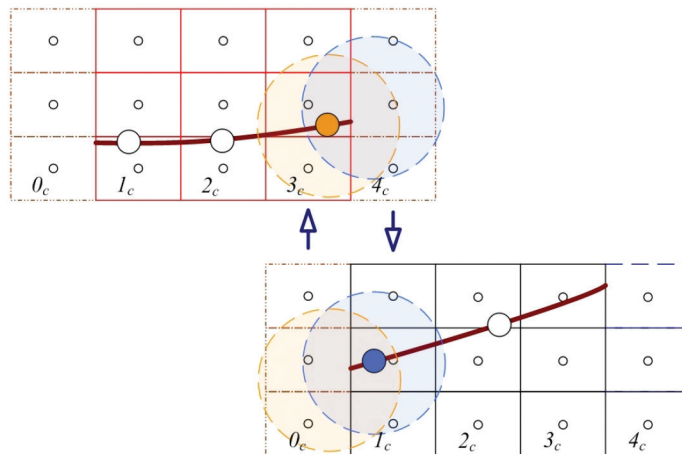
Figure 5 - Correct distribution of Lagrangian forces in a parallel approach. Step 3.

If there is fluid structure interaction, where the immersed object moves across sub-domains, it means that Lagrangian points must be transported across the sub-domains. Since there is no need to retain a explicit connectivity between the triangles that compose the Lagrangian mesh, it is enough to transport information like area of the transported triangle, positions of the normal vector and centroids to another sub-domain. The parallel implementation described above have been fully assessed with good scalability and parallel efficiency up to 83% for 216 processors in a SGI/Altix XE1300 Linux cluster consisting of 30 Intel[®] Xeon[®] processors (26 processors E5650 2.67GHz and 4 E5520 2.27GHz), connected via a Infiniband[®] IB QDR / 10GigE. Since the details of the parallel implementation is not in the scope of the present work, the interested reader may gather more information on the subject in (Vedovoto 2011).

The Immersed Boundary Method as well as the Direct Forcing Method have been implemented in the numerical code FLUIDS3D, (Vedovoto 2011, Vedovoto et al. 2011). The verification of such implementation through the method of manufactured solutions was performed. Moreover, for the sake of validation, two different test cases were simulated and the results will be shown in the next sections.

APPLICATION OF THE METHOD OF MANUFACTURED SOLUTIONS TO THE VERIFICATION OF THE IMMERSED BOUNDARY IMPLEMENTATION

The resort to the method of manufactured solutions is progressively being a well accepted methodology in the framework of numerical code verification (Steinberg and Roache 1985). Such a method consists in developing a priori known analytical solutions of the retained system equations. These manufactured solutions modify the original equations by adding a 'source term', such as those presented in transport equations.

There is an undeniable interest in the use of such a method to quantify accurately numerical capabilities before using computational programs to perform the simulation of more complicated physical systems. This is the case when the immersed boundary is involved. The immersed boundary method creates a perturbation in the flow field in such a manner that it can cause a decay in the rate of convergence of the methods employed.

The evaluation of the Lagrangian force necessary to stop a fluid particle placed at the proximity of an immersed object is dictated by the difference between the actual velocity at a Lagrangian point k and a velocity interpolated from the Eulerian domain at the same Lagrangian point. It is natural, under the method of manufactured solutions, to consider the velocity U_i^{n+1} of Eq. (30), to be the velocity given by the analytical solution.

For the verification of the implemented methods we propose the following analytical solution:

$$u_e = sin^2 \, (\alpha_s \, \pi x + \beta_s \, \pi y + \gamma_s \, \pi z + \delta_s t), \tag{36}$$

$$v_e = -cos^2 \, (\alpha_s \, \pi x + \beta_s \, \pi y + \gamma_s \, \pi z + \delta_s t), \tag{37}$$

$$\omega_e = \frac{\alpha_s}{\gamma_s} \, cos^2 \, (\alpha_s \, \pi x + \beta_s \, \pi y + \gamma_s \, \pi z + \delta_s t) \, \frac{\beta_s}{\gamma_s} \, cos^2 \, (\alpha_s \, \pi x + \beta_s \, \pi y + \gamma_s \, \pi z + \delta_s t), \tag{38}$$

$$p_e = -cos \, (\alpha_s \, \pi x + \beta_s \, \pi y + \gamma_s \, \pi z + \delta_s t). \tag{39}$$

The subscript e stands for the manufactured solutions of the primary variables, i.e. the three velocity components and pressure; t is the time. It is observed that, if the divergence of Eqs. (36)-(38) is taken, the incompressibility condition is verified. The above set of equations is written as a function of the constant parameters α_s, β_s, γ_s and δ_s. Such a set of parameter allow to apply the Method of Manufactured Solutions within a wide range of numerical verifications. For example, if δ_s is set to zero, only the spatial derivatives influence the convergence rate of the scheme. On the other hand, if α_s, β_s, and γ_s are set to zero, only the temporal scheme is verified.

The computational domain considered for the present numerical simulations is a cube of dimensions $[0, 1] \times [0, 1] \times [0, 1]$, in x; y and z directions respectively. The time step is controlled taken the *CFL* factor small than 0.25. The parameters α_s, β_s and γ_s are set to 2, and $\delta_s = 1$. The constant values of density and viscosity are set to unity and 0:01, respectively. Concerning the temporal integration scheme, the Backwards Difference Scheme - BDF, is used.

A sphere of radius 0.25 m, discretized by a surface of triangles, is placed at the position $x = y = z = 0.5$ m. Figure (6) shows the immersed object and its location inside the computational domain. It is important to stress that this is not a physical problem. It is only a matter of verification of the numerical code implementation.

The procedure used for evaluating the error rate of decay is the same presented in Vedovoto et al. (2011), for the evaluation of the global convergence rate of the numerical scheme. From an Eulerian grid of 128^3, the Eulerian grid is halved three times, resulting 4 different grid sizes. The surface mesh of the immersed object follows the Eulerian mesh refinement. As the Eulerian grid becomes coarser or refined, the surface mesh also becomes coarser or refined.

It is important to point-out that in the test presented here, the whole numerical scheme is tested again, i.e, not only the L_2 norm of the immersed boundary must decay, but the rate of the Eulerian scheme must not be affected by the presence of the immersed object. The procedure for the evaluation of both the Eulerian and Lagrangian domains are very similar and were demonstrated in Vedovoto et al. (2011).

The results gathered in the simulations are illustrated in Figure (7). The numerical velocity and pressure are compared with the analytical solutions given by equations (36-39). The inhouse code developed has shown second order of convergence rate for velocity as well as for the immersed boundary. It is an important result since it illustrates one of the main advantages of the direct forcing used in the present work. Since no ad-hoc constants are required in the modeling of the immersed boundary, the
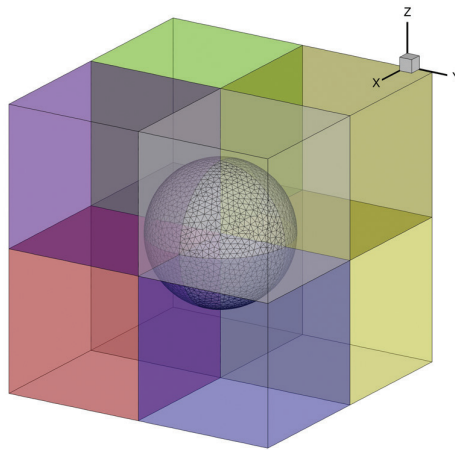
Figure 6 - Immersed geometry in the computational domain. Parallel computing is used. Each one of the Cartesian block is a subdomain assigned to a different CPU or processor core.

convergence rate is dictated by the Eulerian numerical scheme and by the interpolation and distribution functions. Such a characteristic is also observed in high order schemes such as pseudo-spectral based codes, (Mariano et al. 2010).
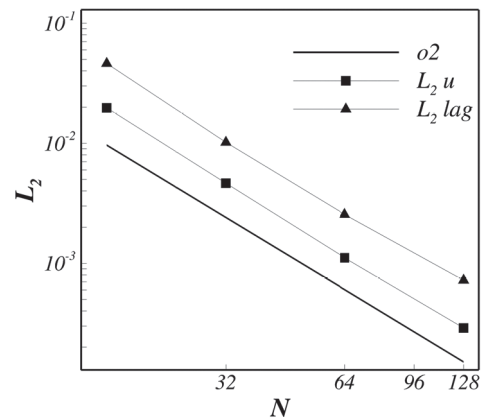

Figure 7 - L_2 norm of the Zero Mach Number manufactured solution. ■: u, ▲: L_2 of the immersed boundary. The solid line stands for the second order decay.

APPLICATIONS

In this section it will be provided a series of applications of the methodologies depicted before.

HIGH VELOCITY TURBULENT JET

In this section a synthetic inlet turbulence generator is presented to perform the numerical simulation of high velocity jets developing spatially. The predictive capability of this methodology is demonstrated via comparisons with experimental data associated to turbulent jet flow.

Jet flows are very dependent on inlet boundary conditions such as mean velocity profile and turbulence intensity and length scale at the nozzle exit. The mean velocity and the RMS profile used in the present work are given in the work of Chen et al. (1996). Only the inert configuration was used in order to perform comparisons. To correctly represent the anisotropic RMS velocity profile of the jet, the method of Smirnov et al. (2001) is used, with the number of Fourier modes set to 1; 000. The turbulent time scale, τ_t, necessary in such a method to evaluate temporal correlations of the velocity fluctuations is set to 1.1 ms (Chen et al. 1996). The computational domain spans a region of $15D \times 15D \times 15D$, in the x, y and z directions respectively. The number of control volumes retained in these respective directions is $225 \times 180 \times 180$, which is approximately the number of control volumes adopted in a previous study of this configuration conducted by Yilmaz et al. (2011). A non-uniform computational grid is used in such a way that approximately 30 control volumes lie near the nozzle, in both the y and z directions, yielding the smallest length of cell size of 0:4 mm. Distributed computing is used and the computational domain is divided into 144 sub-domains, corresponding to 144 processors.

With a Reynolds number $Re = 22400$ the jet can be considered as a highly turbulent jet. The instantaneous field flow is dominated by the presence of high frequency structures, as can be seen in figure (8)-(a). In such a figure it is shown a snapshot of the passive scalar field c at $t^* = tU_0/D = 485$. The filtered instantaneous scalar field once averaged over approximately 10,000 time-steps is reported in figure (8)-(b). The different regions typically found in turbulent jets, such as the potential core, the shear layer and the zone of fully developed turbulence are clearly visible in this figure.
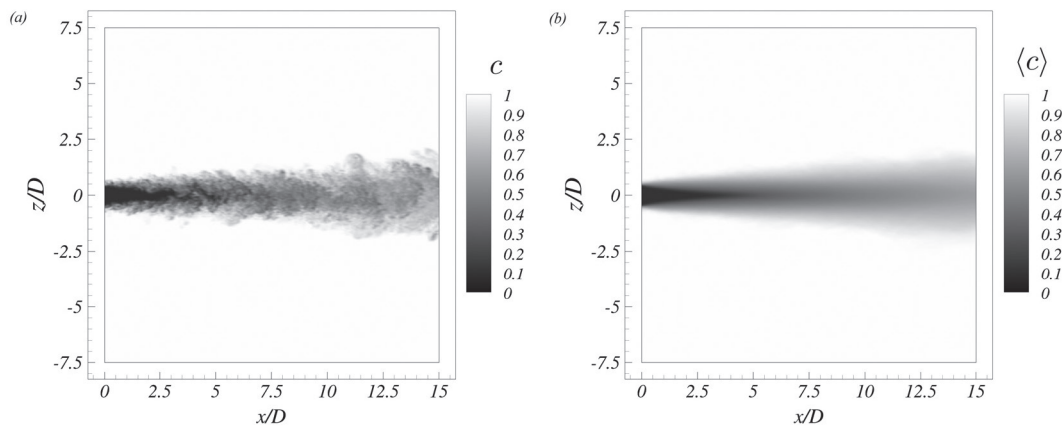
Figure 8 - Scalar variable c for the cold jet: instantaneous (a) and averaged (b) fields.

An evaluation of the length of the potential core is often considered to provide a good indication of the quality of the results. According to Pope (2000) the length of the potential core ranges between 3 and 5 nozzle diameters. In the present case, it is possible to note, with the help of figure (9), that the length of the potential core is approximately *4D*.

In order to quantitatively evaluate this turbulent flow, numerical simulations were conducted with different turbulence models. The Smagorinsky model, with two different values of the constant, $C_s = 0.10$ and $C_s = 0.18$ and the dynamic Smagorinsky model. Figure (9) reports the longitudinal evolution of the mean axial velocity, non-dimensionalized by the mean velocity at the nozzle exit, and the turbulent kinetic energy, non-dimensionalized by the square of the mean velocity at the nozzle exit, along the axial direction

at the centerline of the jet for the three modelings used. The Smagorinsky model, with $C_s = 0{:}18$, is the model that provides the worst results. In this case, it is possible to note that a high value of C_s leads to a reduction of the size of the potential core and to a decrease of the average velocity at the center line that is faster than the measured one. This is followed by an increase in the turbulent kinetic energy, which over predicts the maximum experimental value by almost 300%.

With a smaller value of the Smagorinsky constant, $C_s = 0.10$, there is an improvement in the results agreements with the experimental data. Nevertheless, the unexpected decrease of the average velocity, and the increase in turbulent kinetic energy are still notable. Moreover, the use of $C_s = 0.10$ leads the computational code to present numerical instability issues in this case. In fact, for the results presented in figure (9), obtained with $C_s = 0.10$, even with the resort to a variable time step size, which greatly improves the stability of the numerical code, the simulation is found to diverge after approximately $t^* = 170$, where t^* is the non-dimensional time[1].
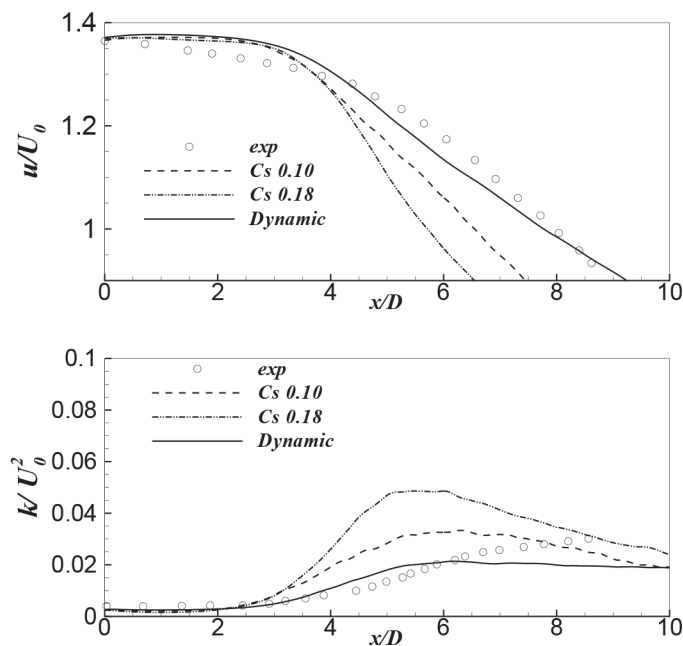
Figure 9 - Streamwise evolution of average velocity and turbulent kinetic energy profiles for the cold jet. The experimental results are extracted from Chen et al. (1996).

The dynamic Smagorinsky model is the turbulence model that captures reasonably well the variations of the average velocity and turbulent kinetic energy along the centerline. The potential core length is well recovered, extending to approximately $x/D = 4.5$, where it is possible to verify that the average velocity decreases as the jet expands along the radial direction. Nonetheless, the turbulent kinetic energy exhibits discrepancies for $x/D > 6$. The reason for that may be due to the fact the computational domain is too short to promote the full development of the jet and hence affecting adversely the developed turbulence zone of the jet.

---

[1]In the present work, a simulation is said to diverge when the size of the time step continuously decreases from several orders of magnitude.

Comparisons of the radial profiles of velocity and turbulent kinetic energy are presented in figure (10). Generally speaking, the results obtained confirm that the code is able to predict the experimental turbulent jet, since the dynamic Smagorinsky closure is used.

Comparing the present results with the numerical results previously obtained by Yilmaz (2008), it is possible to observe that, specially concerning the turbulent kinetic energy, the predictions obtained in the present work are closer to experimental data, even though, in the work of Yilmaz (2008) the flow is also excited in such a manner that the initial profiles of mean velocity and RMS are respected. In their work, the flow is excited by superimposing axial and helical perturbations. Such a technique is usually used in the study of bifurcating jets, see Danaila and Boersma (2000).
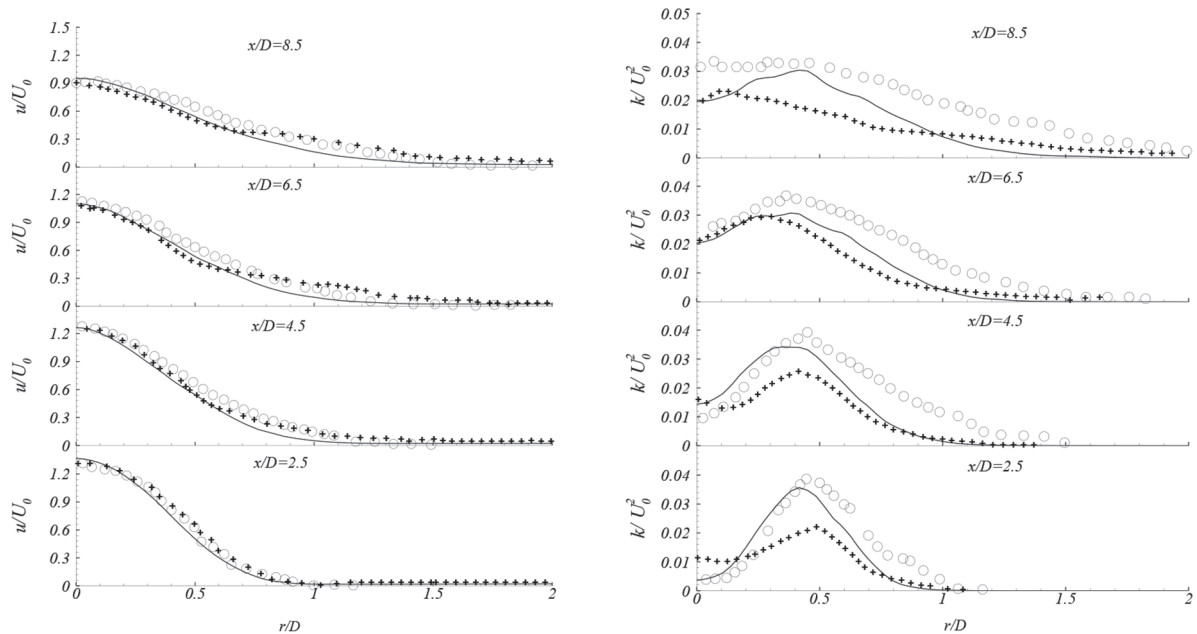
Figure 10 - Radial variation of average velocity and kinetic energy profiles for the cold-flow F3 jet. The solid lines are results of the present work, the plus, (+), symbols are numerical results obtained by Yilmaz (2008) and the circles are experimental results of Chen et al. (1996).

The differences between the obtained radial turbulent kinetic energy profiles (shown in figure 10) and the experimental data, could be explained by the uncertainties in the prescription of the boundary conditions at the inlet. The RFG model (Smirnov 2004), when used for the generation of anisotropic turbulent inflow data requires two different characteristic scales: a length and a time scale. The first, given in the work of Chen et al. (1996) is evaluated from the experimental data. Nevertheless, in the work of Chen et al. (1996) the characteristic time scale at the inlet is given only for the reactive flow cases. For the simulations of the cold-flow presented here, the same turbulent time scale is adopted. Moreover, it is important to note that the turbulent inflow data is generated for the flow issuing from the nozzle only. Elsewhere the velocity fluctuations have been set to zero. In Fig (10) it is clear that for $r/D > 0.5$, i.e., in the outer region of the nozzle, the values of the turbulent kinetic energy are lower than the experimental data. This may be due to the lack of velocity fluctuations calculated in this region.

In these sections two problems are studied. The first one is the Poiseuille flow. The second, a more demanding problem, is the flow over a sphere.

The poiseuille flow

As a part of the validation of the implementation of the Immersed Boundary Method, the Poiseuille flow was chosen since it is one of the few problems in fluid mechanics in which an analytical solution may be provided. The Poiseuille flow is characterized by the developed flow through a long cylindrical pipe, with a constant pressure drop. Figure (11) illustrates the laminar motion of fluid in a horizontal circular pipe located at a sufficiently great distance from the entrance section in order to have a developed flow. It makes a good test for the Immersed Boundary Method since, although geometrically simple, the tube itself must be contained into the computational domain, which is Cartesian. Therefore, the whole tube must be modeled with the Immersed Boundary Method.
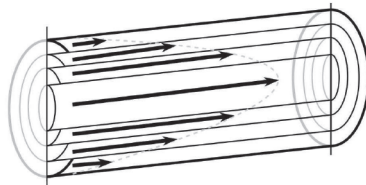
Figure 11 - Cross section showing the velocity vectors in a Poiseuille flow.

The analytical velocity profile at any position across a section of the tube can be given by Eq. (40):

$$u(r) = \frac{-1}{4\,\mu}\frac{dP}{dx}(R^2 - r^2) \tag{40}$$

where, μ is the kinematic viscosity of the fluid, R is the radius of the tube, and dP/dx is the pressure gradient, which must be given and in the present case it is set to 1.

Figure (12) shows the computational domain, and the immersed geometry, as well as the u-component of velocity across two sections. It can be noted that due to imposed pressure drop only within the tube, the external flow has no influence on the internal flow. The computational domain is a box of dimensions $1 \times 0.5 \times 0.5$ meters, discretized in a $200 \times 100 \times 100$ control volumes in x, y and z directions respectively. The Reynolds number, based on the diameter of the tube $\phi = 0.4$ m, and the maximum value of the velocity component u, determined by the analytical solution, 5 m/s, is $Re = 100$. Periodic boundary conditions are used in the x direction, while Neummann boundary conditions are set in y and z directions. It worthy noting that these boundary conditions are set at the rectangular Eulerian domain. For the immersed object, non-slip conditions are adopted for the tube imposed using the immersed boundary method.

In order to provide a quantitative comparison, figure (13) shows the result of a probe positioned along the z-axis, y = 0.25 m and x = 0.5 m. The maximum difference obtained was less than 1%. Hence, the implementation of the Immersed Boundary Method for the Poiseuille flow can be considered validated.
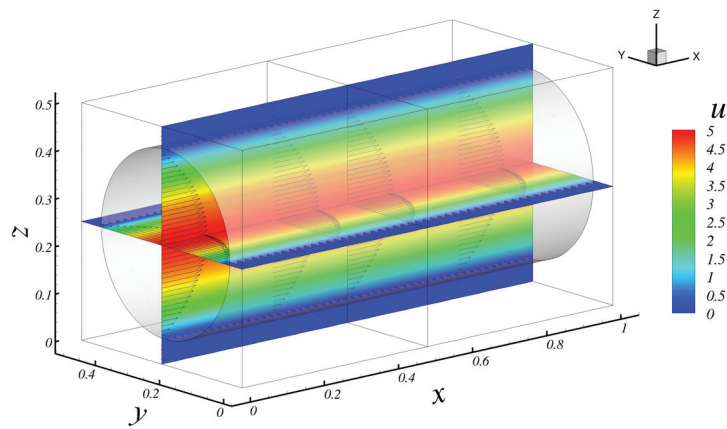
Figure 12 - Poiseuille flow. Computational domain and u-component of the velocity.
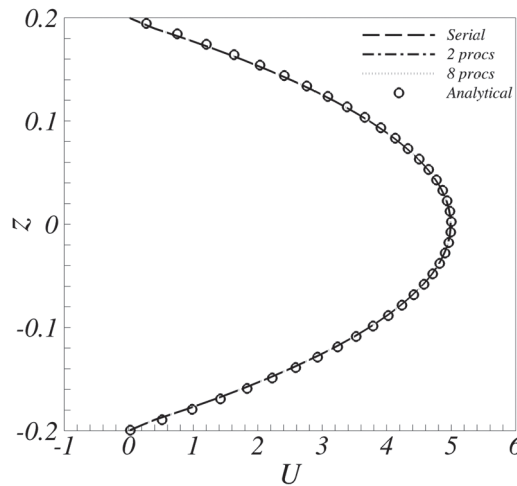

Figure 13 - Quantitative comparison of the velocity profile for a circular Poiseuille flow, obtained with the Immersed Boundary Method against an analytical solution.

The flow over a sphere

In this section, the results of flows over a sphere will be presented. The immersed boundary methodology uses one Eulerian domain for the fluid as a whole and one Lagrangian domain for the immersed boundary. These two domains relates each other by interpolation and distribution process. The Lagrangian force field is distributed over the fluid Eulerian grid and the Eulerian velocity that results from the Navier-Stokes equations solution is interpolated to the Lagrangian domain. The Eulerian grid is Cartesian, regular and its generation is straightforward. The Lagrangian grid must describe surface of the body over which the flow must happen. It is generated using a triangular mesh that describe its surface. The FLUIDS3D numerical code, developed in the MFLab laboratory, imports a stl extension file in a very simple and practical way (Vedovoto 2007).

Figure (14) shows both Eulerian and Lagrangian meshes. The Lagrangian mesh is immersed in the Eulerian mesh. The Eulerian mesh can be non-uniform in such a way that it is possible the refine locally over the immersed boundary. A mesh of 2.366.000 volumes was used in the present work. This figure shows the two mesh volumes, the Eulerian volume, $dV = h^3$, where h is the mesh size, and the Lagrangian volume $dV = (4\pi r^2 h)/Nl$, where r is the sphere radio and N_l is the number of elements that compose the surface.

In order to determine the number of elements, it is supposed that the Lagrangian element has approximately the same volume as the Eulerian element, as proposed by Uhlmann (2005). This way, the number of Lagrangian elements can be determined as $N_l = 4\pi r^2/h2$. For the present work, the sphere diameter is taken as $d = 0.04$ m and 1257 Lagrangian elements were used.
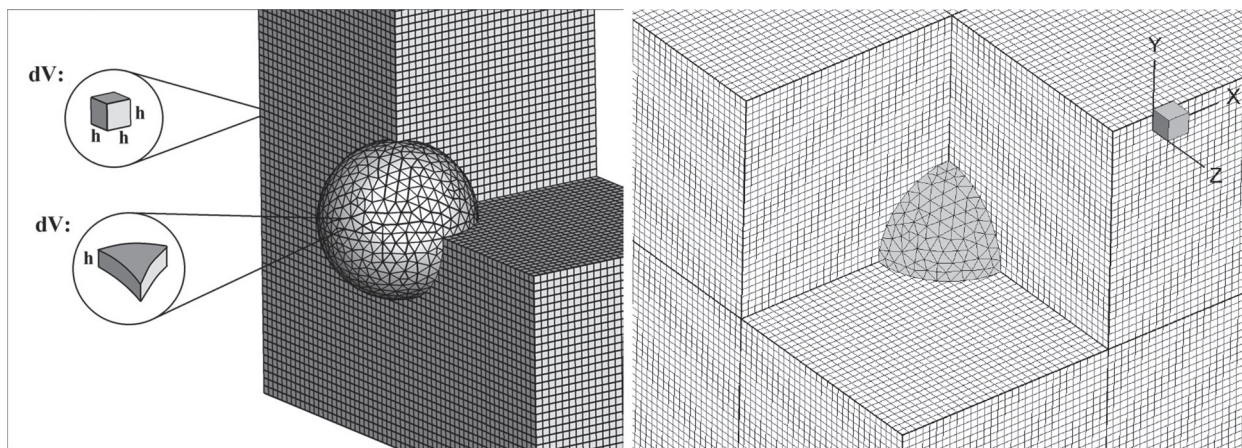
Figure 14 - Eulerian and Lagrangian mesh and control volumes.

The initial conditions were taken as the domain inlet velocity in the x direction. The Dirichlet boundary condition was imposed at the inlet domain and Neumann boundary conditions applied elsewhere. The Reynolds number was calculated using the inlet velocity U_1, the sphere diameter d and the cinematic viscosity v of the fluid, $Re_d = Ud/v$. Several values of the Reynolds number were simulated, ranging from $Re_d = 10$ up to $Re_d = 10^6$ in such a manner that the simulated flows range from laminar to high turbulent regimes. The simulations were performed for a time $t^* = U_{in}t/D = 300$. The statistics were assessed for a period $t^* = 200$, or approximately eight flow through.

Figure (15) shows the streamlines superposed to the velocity fields. The results for the flow under three different Reynolds number are shown. The immersed boundary represented well the physical boundary condition over the sphere and the classical recirculation was captured.

The higher is the Reynolds number, as expected, more unstable is the flow. The Q criterion (Jeong and Hussain 1995), which identifies regions in the computational domain with high rotational characteristics, is used in order to visualize coherent structures in the flow, as can be seen in figure (16). We can visualize in such figures another remarkable characteristic of these kind of flows: it becomes more turbulent as the Reynolds number increases evidencing even longitudinal filaments known as hairpin structures.
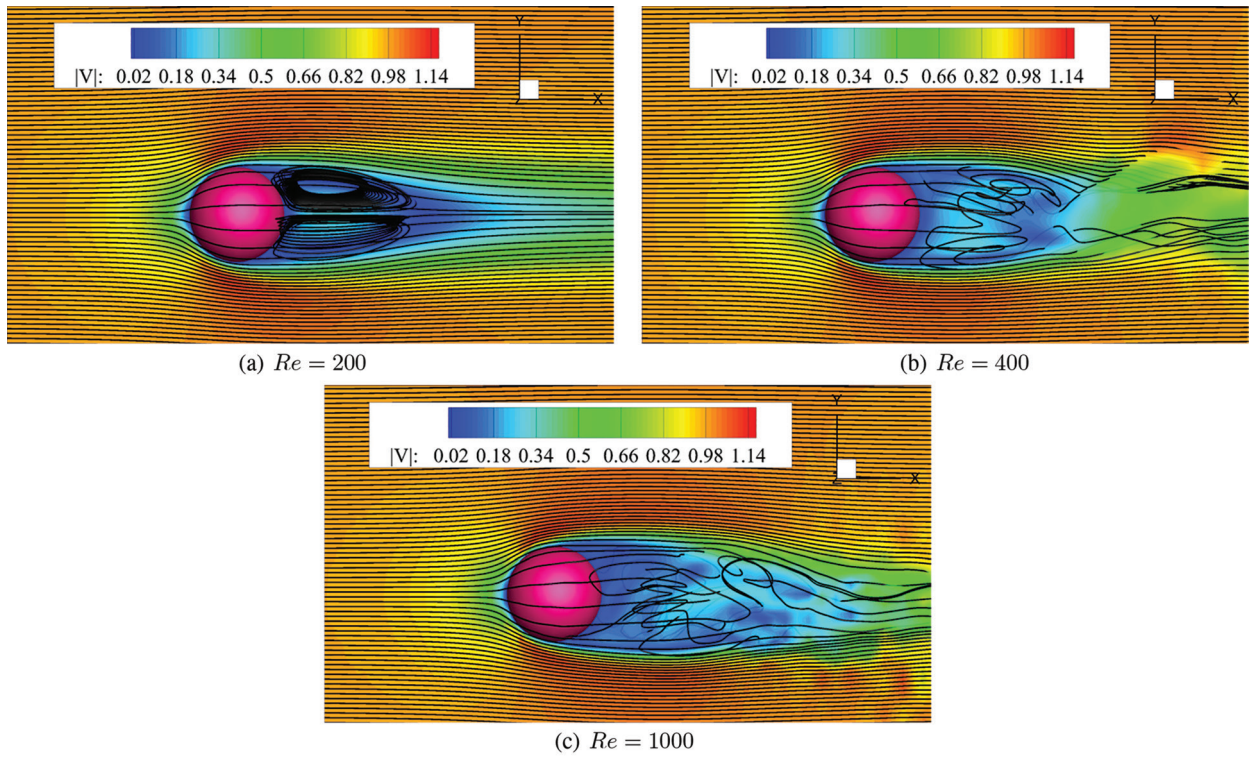
Figure 15 - Velocity field and streamlines for the flow over the sphere, for three values of the Reynolds numbers: **(a)** Re=200, **(b)** Re=400 and **(c)** Re=1000.
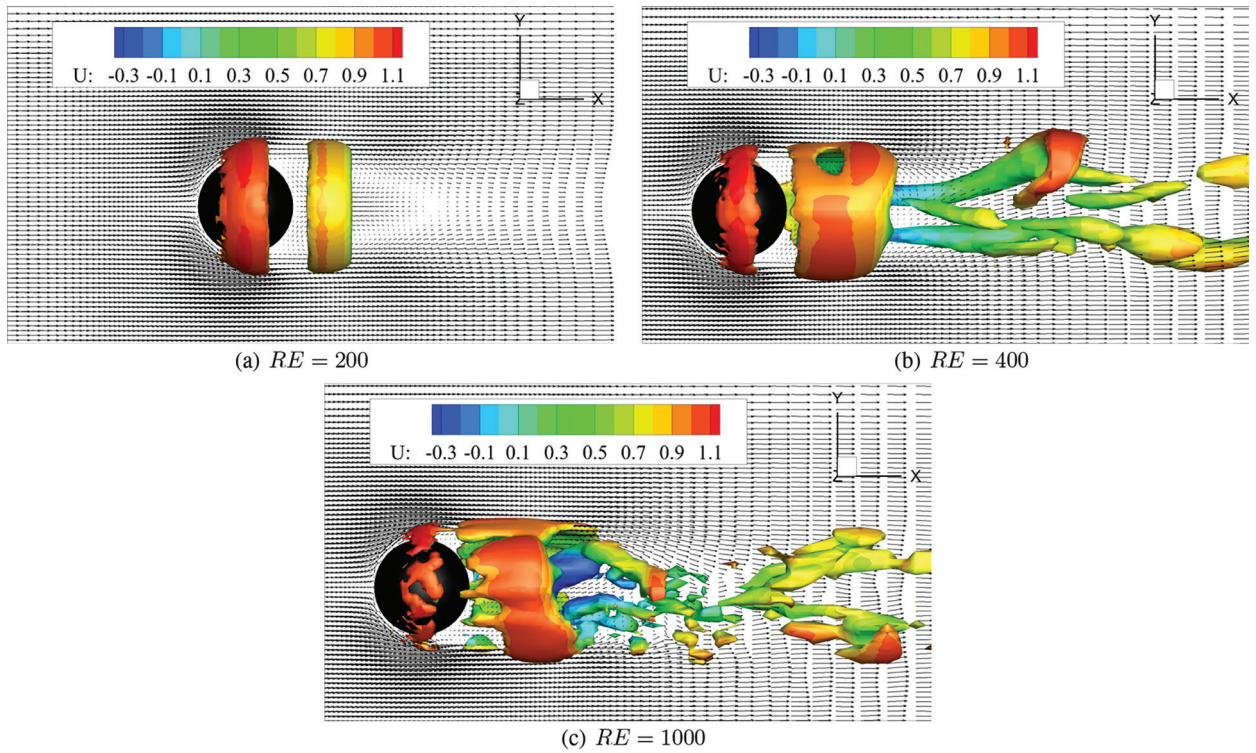

Figure 16 - Velocity vectors and isovalue of Q=300 for three distinct Reynolds numbers: **(a)** Re=200, **(b)** Re=400 and **(c)** Re=1000.

In order to assess the boundary condition representation by the immersed boundary methodology, the L_2 norm, which is defined by the following equation, is used

$$L_2 = \sqrt{\frac{\sum\limits_{k=1}^{NPL} \| NS_k - EX_k \|}{NLP}} \qquad (41)$$

where NLP are the total number of Lagrangian points representing the immersed boundary, NS_k is the numerical solution and EX_k is the expected solution, in this case, EX_k stands for the imposed velocity at a Lagrangian point k. In the present work, the L_2 norm for the velocity at the immersed boundary was evaluated, where the calculated velocity at the interface was obtained by the interpolation of the Eulerian velocity at the neighboring of the interface and the expected value of the velocity at the interface is zero, since the sphere is considered not moving. This parameter was calculated for several Reynolds number and the result is shown in figure (17), where acceptable values, that guarantees non-slip condition at the immersed interface, were obtained.
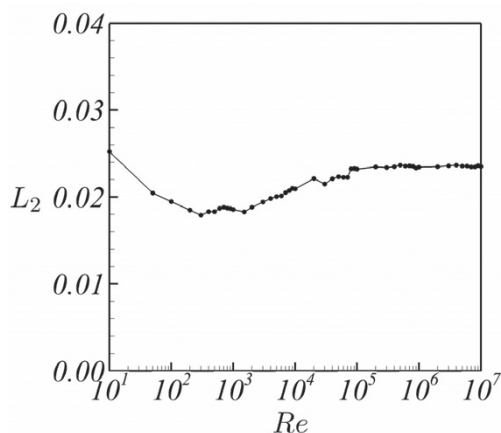
Figure 17 - L_2 norm of the velocity at the immersed boundary for different Reynolds numbers.

The drag force is calculated taken the projection of the force vector, over the flow direction, on each Lagrangian volume. The drag coefficient is defined as

$$C_D = \frac{\sum\limits_{k} F_D\left(\vec{X}_k\right)}{\frac{1}{2}\rho U_{in}^2 \left(\pi \, d^2/4\right)} \qquad (42)$$

where $F_D\left(\vec{X}_k\right)$ is the drag force that actuates over each Lagrangian point $\vec{X}_k$.

Figure (18) presents the time evolution of the drag coefficient for four Reynolds number values. The drag coefficient decreases as the Reynolds number increase. Time instabilities over this parameter appears for high Reynolds number due to the physical instabilities that appears downwards the sphere.

Figure (19) presents the drag coefficient as a function of the Reynolds number, ranging from Re = 10 to Re = 1000. The results of the present work are compared with the experimental results of Fornberg (1988) and numerical results of Campregher (2005) and Subramanian (2003). The correlation of Subramanian (2003) is given by the following equation:
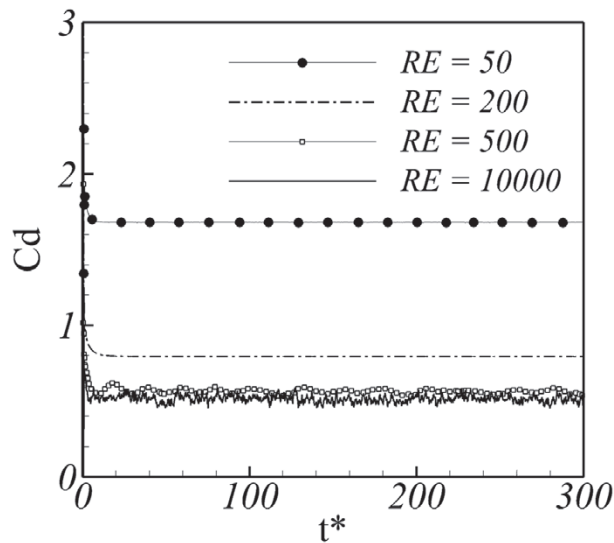
Figure 18 - Time distribution of the drag coefficient for four values of the Reynolds number.
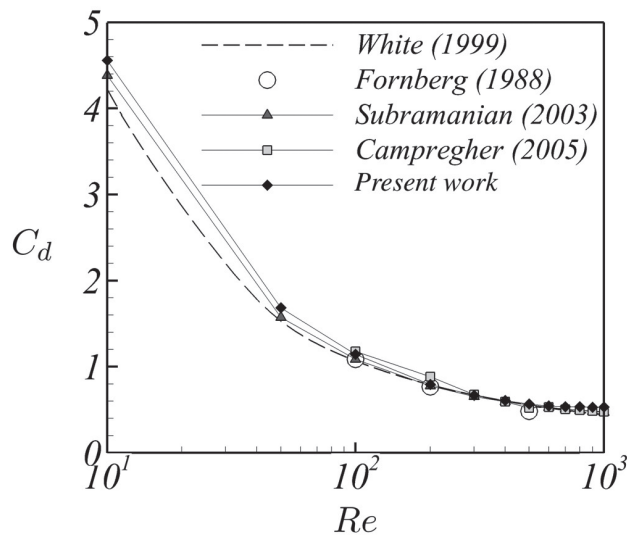

Figure 19 - Drag coefficient as a function of the Reynolds number: $Re = 10$ up to $Re = 10^3$ (Fornberg 1988, White 1999, Subramanian 2003, Campegher 2005).

$$C_D = \frac{24}{Re} \left[1 + 0.1935 \left(Re^{0.6035} \right) \right] \qquad 20 \leq Re \leq 260$$
$$C_D = 0.16435 - 1.1242 \left(\log_{10} Re \right) + 0.1558 \left(\log_{10} Re \right)^2 \quad 260 \leq Re \leq 1.5 \cdot 10^3 \qquad (43)$$

The drag coefficient for the sphere was also evaluated for high Reynolds numbers. This parameter, ranging from $Re = 10$ up to $Re = 10^6$ is presented in figure (20). The results agree well with experiment, except that the drag crises could not be captured in the present work. The drag crisis results of the boundary

layer transition to turbulence upwards of the boundary layer detachment line. The numerical method and the level of the grid refinement near the sphere are not adequate to simulate the very fine instabilities that characterize the boundary layer transition.
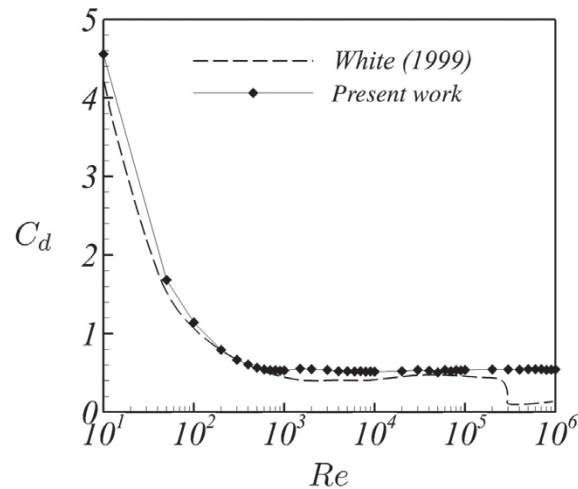
Figure 20 - Drag coefficient as a function of the Reynolds number: $Re = 10$ up to $Re = 10^6$; present work and White (1999).

Flow inside an industrial fan

An industrial fan was simulated in order to illustrate the potential of analysis of the methodology presented, verified and validated in the present work. Here, we simulate two different geometries of the rotor of an industrial fan. In this case the CFD analysis is fundamental to verify if a proposed modification may affect the overall functional parameters, like mass flow rate and the pressure drop measured at the inlet and outlet of the equipment.
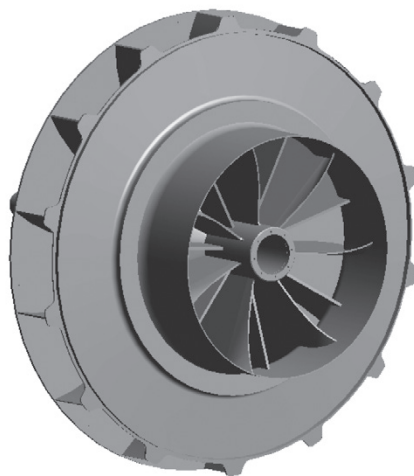
Figure 21 - Original rotor design.

As afore mentioned, the Immersed Boundary requires two different independent domains, i.e., an Eulerian domain, where the differential transport equations are solved, and a Lagrangian domain, representing an immersed geometry in the computational domain. It was also commented that the immersed boundary is composed by a cloud of Lagrangian points where equation (30) must be evaluated in order to enforce a boundary condition. This implies that the immersed geometry may be represented by an superficial mesh, as represented by any stl file. We make use of this facility to simplify an immersed geometry like the rotor presented in figure (21), which is the actual rotor of our simulated fan, and with no loss of quality, to minimize the width of the steel plates and to represent the rotor blades by planes, as depicted in figure (22). In this figure we show the fan external frame, also composed by a triangular grid and a zoom view of the fan inlet and its 14 small blades. It is worth noting that such a rotor have a highly complex design, with 14 blades at the inlet and 7 bended blades that are equally positioned along the rotor.
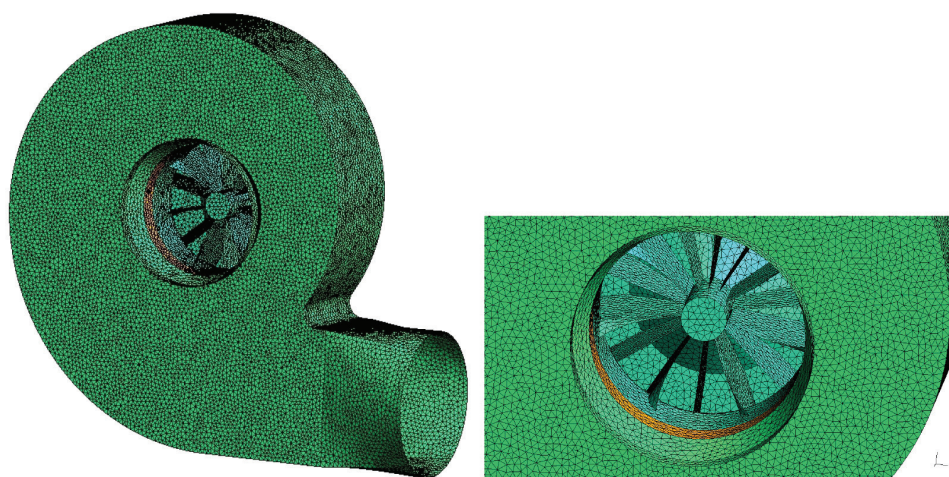
Figure 22 - Three-dimensional views of the fan; surface triangular mesh used.

The idea of a CFD analysis comes from a desired modification in the diameter of the rotor. Originally, in its highest length, the rotor diameter has 2.5 meters. It is proposed to decrease such diameter in 80 mm in order to diminish the fluid-dynamic stresses imposed on the equipment. On the other hand, such modification should not affect the main characteristics of fan, so, we simulate, under operational conditions, the original and modified rotors and assess the results aiming at to demonstrate that the modification proposed has an important impact in the fluid-dynamic stresses imposed on the rotor, but little impact on the pressure at the outlet for instance. The difference between the modified (presented in green) and original (yellow) rotor is evidenced in figure (23).

The transient simulations were carried out under the following conditions:

Volumetric flow rate: 60 *Tons/h* (16 Nm^3/s) of air at 69.2 °C;
Operational velocity: 1650 rpm;
Specific weight at the given temperature: $\rho = 0.5842$ *kg/m*^{3};
dynamic viscosity at the given temperature: $\mu = 8.659 \times 10^{-6}$ *Pa · s*;
Resulting Reynolds number at the inlet: ≈ 2.2 millions;
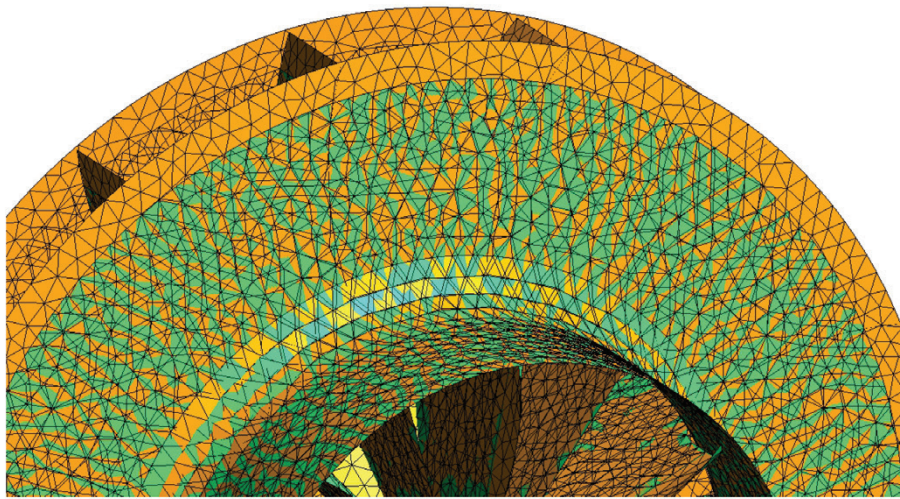Resulting Reynolds number at the rotor tip: ≈ 36.85 × 10^6;

Figure 23 - Original (yellow) and modified (green) rotor design.

With the transient simulations and the resort of turbulence modeling by means of Large Eddy Simulations we may quantitatively differentiate the results of the both rotors and actually to provide a more important comparison data: we are capable of identify the main frequencies generated by the fluid flow and if any of these characteristics frequencies may incur in the resonance phenomena. It is noteworthy that besides DNS, that still lies far from being feasible for a simulation of a real problem like the presented, LES is the only methodology capable of recover detailed informations. Until a few years ago, to perform LES on such a high Reynolds flows, in a such complex geometry like the presented was unthinkable.

Since the velocity of the rotor is imposed by an electric motor at constant angular velocity, and due the high speed, it is natural to expect that with only a small reduction in the diameter of the rotor the main characteristics of the velocity field would not alter. We can see a few of the characteristics in figure (24). In this figure the results of the averaged velocity and pressure fields are shown over a plane positioned in the plane XY, in the interior of the rotor, for the original geometry. Streamlines are plotted indicating the track followed by a fluid particle entering the inlet, passing by the first part of the rotor, and being impelled by the rotor. We can note that there is a strong rotation of the fluid leaving the rotor towards the outlet. Two interesting characteristics may be cited: the first one is the very low relative value of pressure at the center of the rotor indicating a strong acceleration of the fluid, but also indicating strong stresses imposed on the rotor[2]. The second notable characteristic is the high velocity of the fluid only in the outer regions of the rotor, with a rapid decrease in the velocity when the fluid is moved towards the outlet.

The simulations were carried out using a computational domain with dimension of $5 \times 5 \times 1$ [m] in the directions x, y and z respectively. This domain was discretized in a Cartesian uniform grid of $200 \times 200 \times 100$ control volumes, summarizing a grid of 4 million control volumes. Since the computational code developed is capable of distributed computing, the computational domain was divided 100 sub-domains, allowing a feasible CPU time cost for a LES simulations of this magnitude. The size of the time step is

---

[2] We may recall that this pressure variation acting on a surface will impose a fluctuating force on the structure with a characteristic frequency. If this frequency is the same of any of the natural frequencies of the structure we may have the development of cracks or even catastrophic failure.
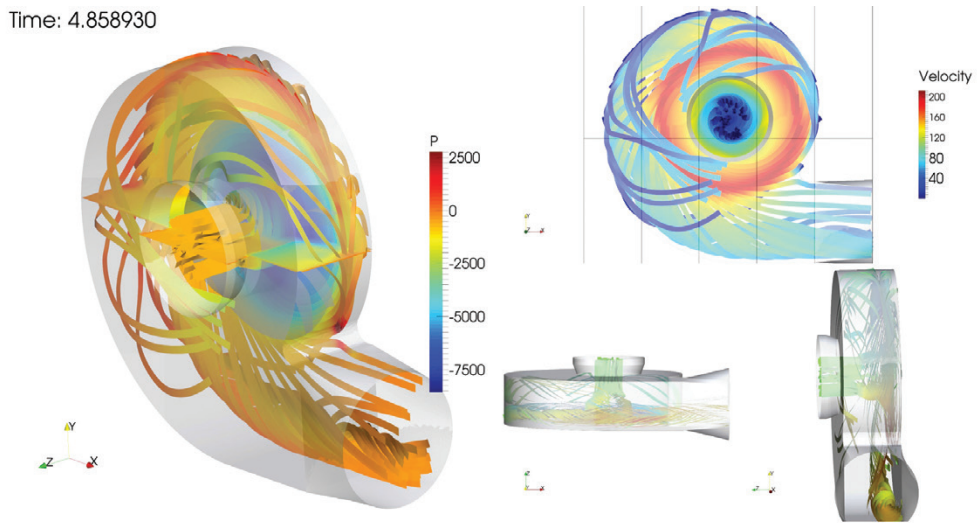
Time: 4.858930

Figure 24 - Mean pressure and mean velocity fields magnitude.

controlled by the CFL criteria and was set to 0.5, what leads to an average value of time step of $3.0^{-5}s$. In order to evaluate the mean value of the variables of interest, the instantaneous fields were averaged approximately 40.000 times, meaning a regime statistically permanent[3].
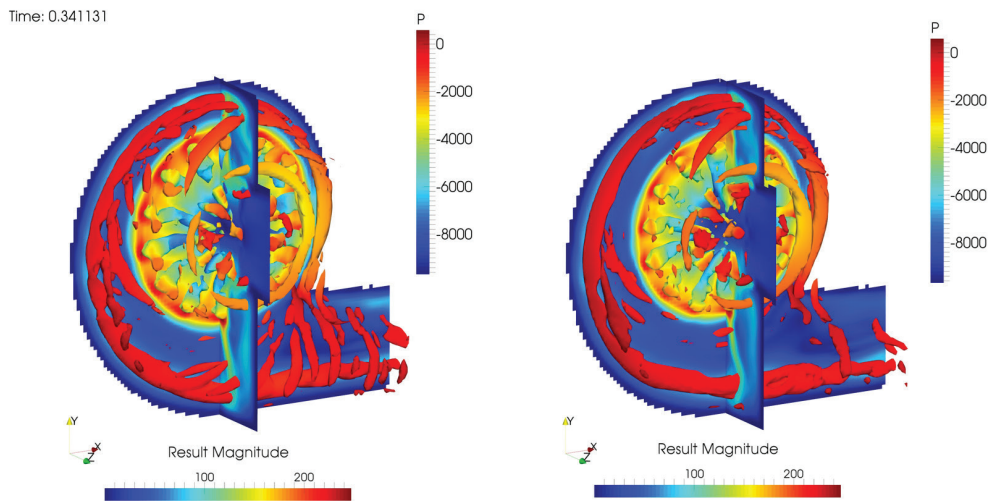

Figure 25 - Instantaneous field of velocity magnitude and isosurfaces of the criteria $Q = 10^6$ at $t = 0:34$s. Left: original rotor; right: modified rotor.

We may compare the differences between the simulations with the original and modified rotors by analyzing isovalues of the Q criteria (Jeong and Hussain 1995), that basically dictates regions in the computational domain where the fluid dynamics is predominantly rotational. In figure (25) we show the

---

[3] It is important to keep in mind that this flow is not permanent and it will not achieve a permanent regime, however, we may achieve a regime statistically permanent.

isosurfaces of $Q = 1^6$ for both original and modified rotors, colored by pressure, along with a slice positioned at the plane XY at the center of the rotor showing an instantaneous field of velocity magnitude. As stressed before, there is no notable change in the overall behavior of the velocity field, however, when reducing the diameter of the rotor, we have noted a decrease in the presence of vortical structures near the fan external frame. This must be due to the slight reduction on the velocity magnitude of the flow. In the next section we will assess this effect by performing a signal analysis.
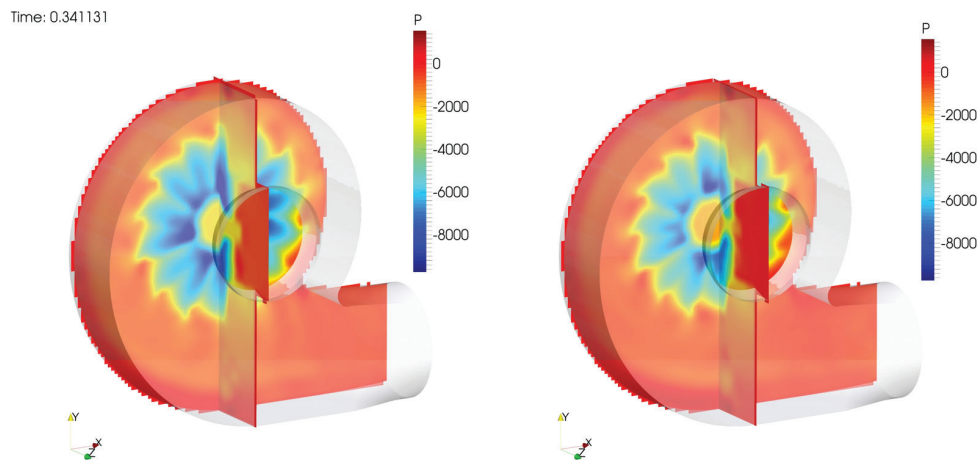
Figure 26 - Instantaneous field of pressure at $t = 0.34s$; left: original rotor; right: modified rotor.

The main difference between the flows with the original and modified rotors are connected to the variations on the pressure field. In figure (26) it is evidenced that even in an instantaneous view, the pressure reaches higher values in the original rotor, indicating that the structure may be subject to higher stresses. Such a behavior is evidenced also in the figure (27) where we show the pressure field averaged over approximately 15000 samples, meaning that the pressure drop evidenced in instantaneous snapshots are present along time.
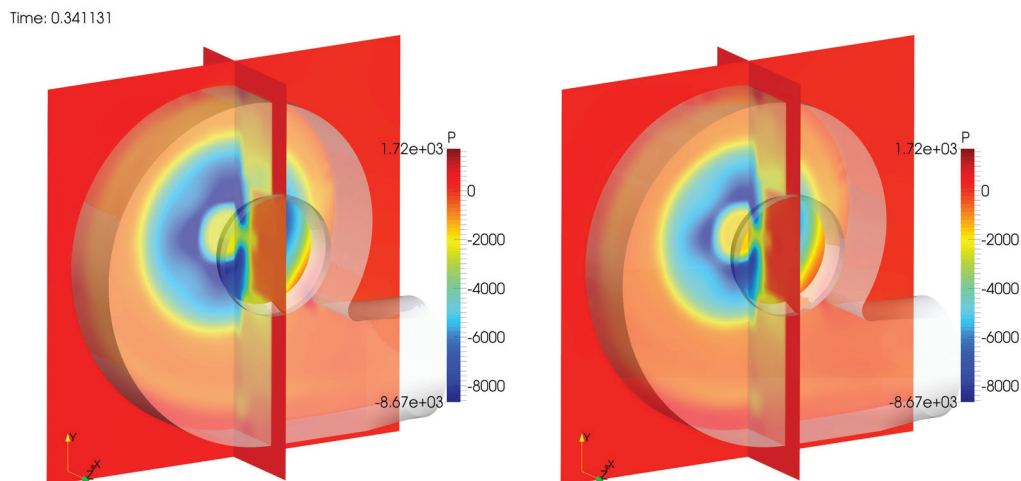

Figure 27 - Averaged field of pressure; left: original rotor; right: modified rotor.

We use figure (27) to explain another characteristic of the immersed boundary methodology. It is possible to verify that the plan crossing the rotor also crosses the limits of the fan frame. It indicates that there is, indeed, an evaluated flow field outside the fan frame. One may question if such regions would not result only in an increased computational cost. Indeed, it may increase the overall computational cost, however, a great advantage is obtained with a reasonable grid resolution outside the domain of interest. Since the Immersed Boundary Method is based on a force field evaluation between a set of Lagrangian points, the complementary flow field generated outwards the fan frame helps to ensure the no-slip boundary condition on the immersed object.
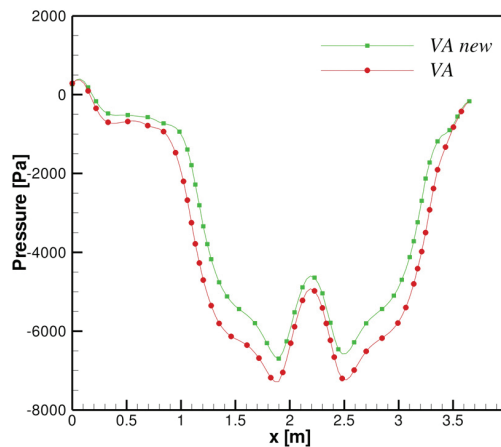
Figure 28 - Variations in the mean pressure along the rotor for the original and modified geometries.
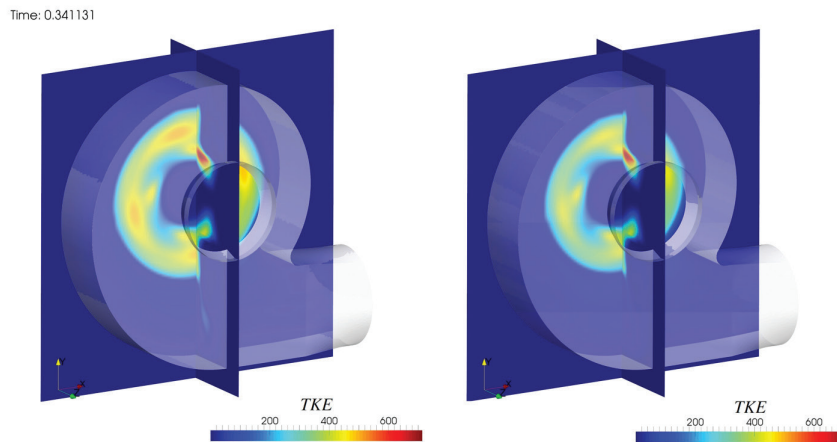

Figure 29 - Averaged field of turbulent kinetic energy; left: original rotor; right: modified rotor.

In order to assess quantitatively the effects of modifying the rotor diameter on the pressure, in figure (28) it is shown the results of a linear probe positioned across the fan (passing through the rotor) where the average pressure is measured. We can clearly see that the pressure drop in the original rotor is about 1kPa higher than its correlate modified rotor. We see the result of these less intense pressure variations on the turbulent kinetic energy (TKE) $\bar{k} = 0{:}5(\overline{u'u'} + \overline{v'v'} + \overline{w'w'})$, displayed in figure (29).

SIGNAL ANALYSIS

In this section, we show results of a signal analysis over temporal-extracted probes positioned in a few important places in the computational domain. With these probes it is possible to extract insightful data about the unsteady behavior of the flow, as well as to identify possible frequencies of vortex shedding that may be harmful to the equipment.
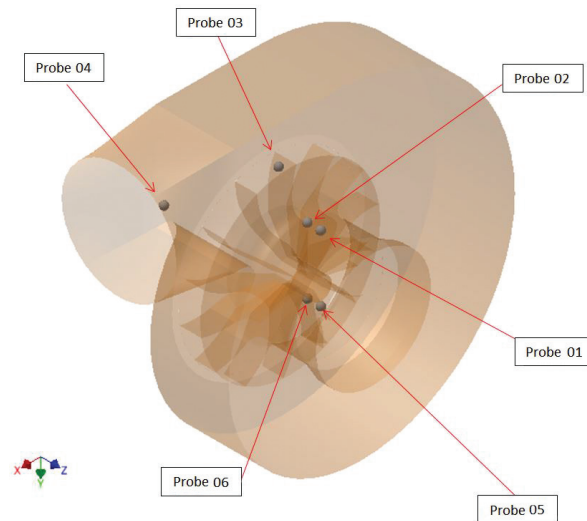

Figure 30 - Positions of the numerical probes in the computational domain.

The probes are positioned accordingly figure (30). There are six, strategically positioned at the coordinates x; y; z = (2.534 × 2.181 × 1.311), (2.534 × 2.181 × 1.136), (2.534 × 1.741 × 0.761), (4.399 × 1.056 × 1.128), (2.534 × 3.041 × 1.311), (2.534 × 3.041 × 1.136) named probes 01 to 06 respectively. Probe 01 is positioned at the tip of the small blade near de inlet of the fan. Probe 02 is located at the junction of the small blade to the large blade, probe 03 is at the tip of the large blade, probe 04 is near the outlet, Probe 05 and 06 are at near the shaft, but respectively at the beginning of the rotor and at the center of the large rotor.

It is important to point out that all the energy spectra showed in the manuscript were obtained using temporal series, and measured at discrete coordinates. The samples (discrete values of turbulent kinetic energy) were taken in such a manner that they are equally time-spaced, hence allowing the use of the discrete Fast Fourier Transform to identify the characteristic frequency of eddies, f, as indicated in the abscissa of figures (31-36).

It is usual in the turbulence literature to plot the energy spectra in log scale since the it become more clear the wide energy distribution along the multiples scales, however, in the present work we also show the energy spectra in linear scale in order to evidence a remarkable feature of this kind of flow concerning the evolution of turbulence.

One of the advantages of Large Eddy Simulations is to capture unsteady phenomena. Given the rotor's velocity of 1650 RPM and the number of blades, seven large and fourteen small blades, we can find the blade passing frequencies, 192.5 Hz and 384 Hz respectively. In figure (31) it is displayed the temporal

energy spectra of probe 01. In the bottom part of the figure, there is a region of −5/3 as expected from a fully turbulent flow by Kolmogorov's theory, however, one interesting fact is that the results also show some specific frequencies like 192.5 and 384 *Hz* as well as its harmonics, matching the blade passing frequency. This fact only is noteworthy since the numerical framework must be prepared to capture such flow characteristics. Concerning the effects of the change in the rotor diameter, for probes 01 and 02, little can be said, since the flow in these regions are basically the same. The energy spectra associated to probe 02 is displayed in figure (32).
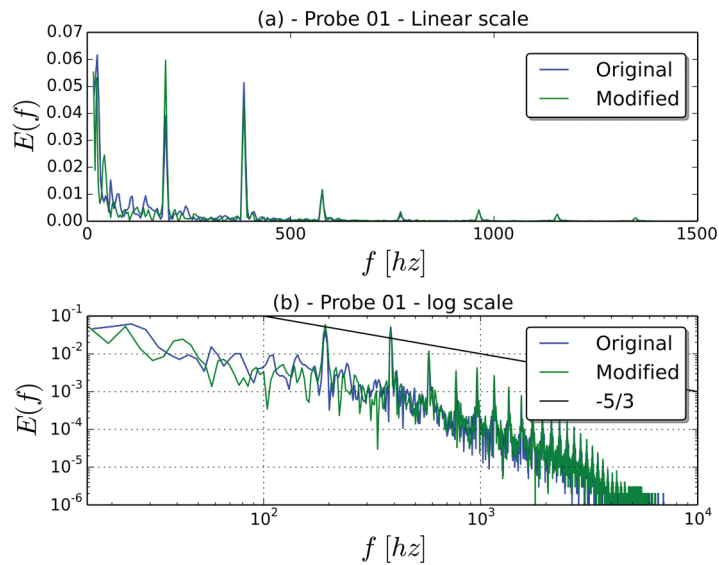
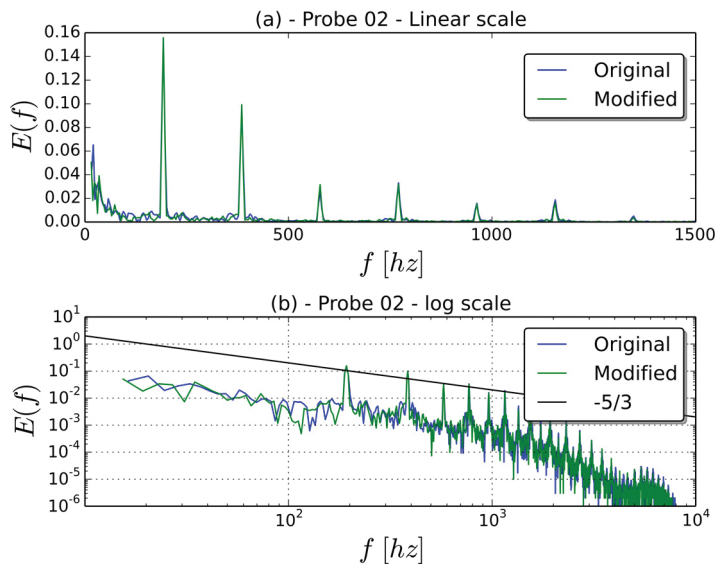**Figure 31 -** Energy spectra for probe **1**. Left: in linear scale; right: log scale.

Figure 32 - Energy spectra for probe **2**; **(a)**: in linear scale; **(b)**: log scale.

The blade passing frequencies as well as its harmonics are felt by the fluid and can be clearly noted in figure (33). Since this probe is positioned in such away that even with the reduction in rotor diameter the blades will pass trough it, little change is found in the turbulent kinetic energy.
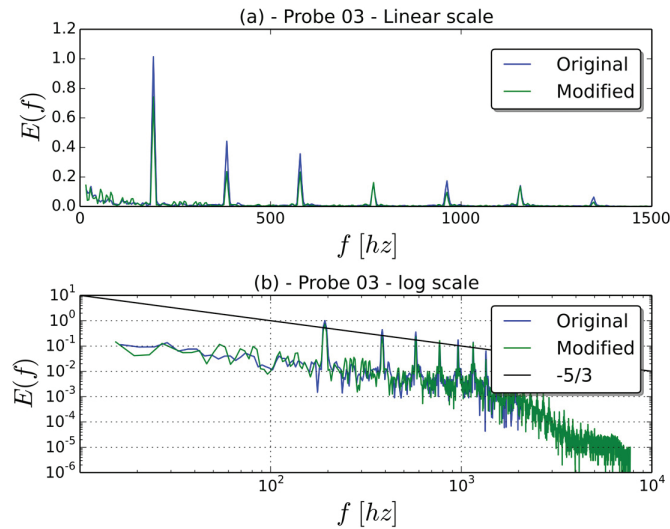

Figure 33 - Energy spectra for probe 3; **(a)**: in linear scale; **(b)**: log scale.

The effects of the geometry modification is notable in the energy spectra obtained from figure (34). Although the frequencies match for the original and modified rotors, it is possible to verify a reduction in the energy associated the frequencies in the time of the blade passing. This energy reduction on the scales may be associated to the decrease of the pressure drop in this region. Since probe 06 is positioned further in the fan frame, the energy transfered from the rotor to the fluid is more pronounced, and hence, again, we may not note a great difference before and after the modification in the rotor's geometry, as displayed in figure (35).
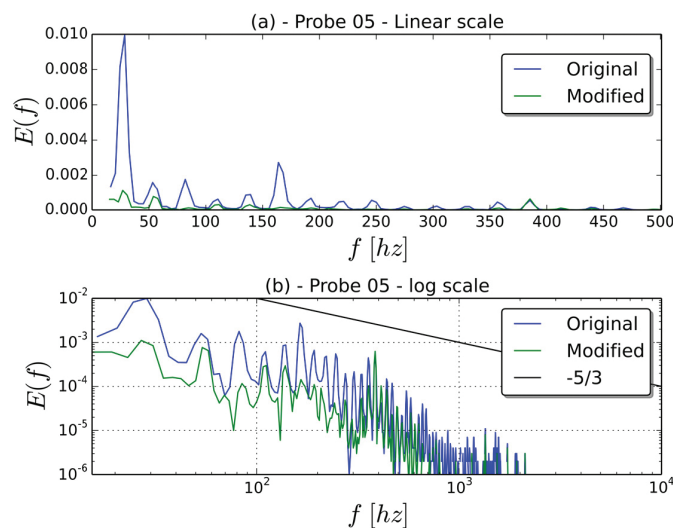

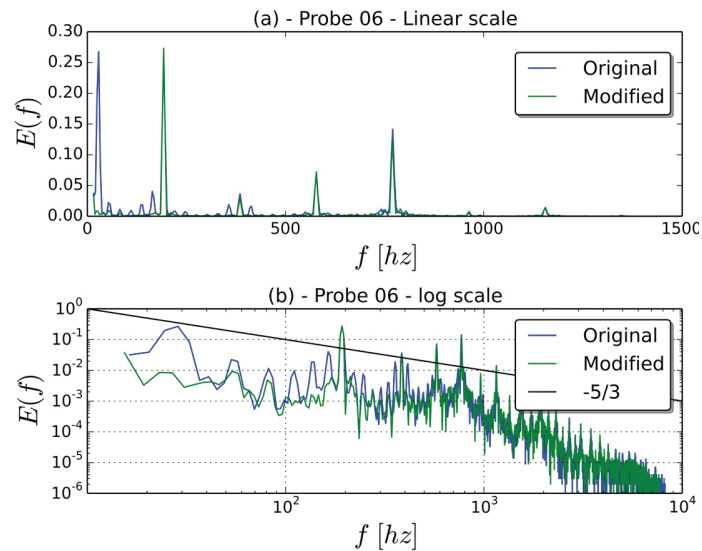**Figure 34 -** Energy spectra for probe 5; **(a)**: in linear scale; **(b)**: log scale.

Figure 35 - Energy spectra for probe 6; **(a)**: in linear scale; **(b)**: log scale.

Finally we assess the results of probe 04. It is important to point-out that this probe is positioned far from the rotor, but still in a region where the fluid behavior is strongly anisotropic and not fully developed. For this reason, we may not expect a well defined −5/3 region in figure (36). However, we note the frequencies associated to the blade passing are vanished. We also infer from figure (36) a slight decrease on the energy associated to turbulent scales when the rotor diameter is reduced.
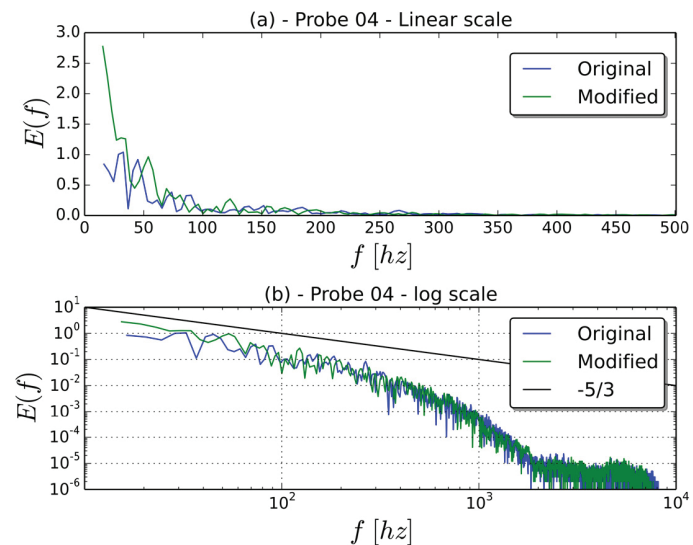

Figure 36 - Energy spectra for probe 4; **(a)**: in linear scale; **(b)**: log scale.

The main objective of this section was to briefly present an industrial application of LES methodology. In this case it was important to verify that a modification in a key part of an industrial equipment would result in beneficial operational condition, without resorting to tentative and error procedures. The authors consider that the obtained results represent very well the chosen industrial problem. The solution of the fluid

dynamics inside the fan furnishes the instantaneous pressure fields which can be used to solve the structural problem and prevent vibration induced by turbulence. We have shown that even in a application involving a Reynolds number of order of millions it is possible to have an acceptable answer, provided a numerical framework suitable for LES.

CONCLUSION

In this work we presented an overview on the Large Eddy Simulation, the mathematical and numerical requirements in order to have an acceptable result. Important features like the pressure-velocity coupling, discretization of transport equations, grid choice, parallelization leading to feasible computational cost are some of the characteristics discussed above. In order to simulate complex geometries on Cartesian grids the Immersed Boundary Method was presented, as well as its key functionalities that must be respected even in distributed environment. An in-house developed numerical code (FLUIDS3D) was successfully used, where care was devoted to the verification and validation of each method discussed here, and once the numerical framework was fully verified and validated we were able to simulate key flows like turbulent jets, internal flows, the turbulent flow over a sphere, and an industrial application of the flow on a high velocity fan. For the last one, it was demonstrated that it is possible to take engineering decisions based on CFD analysis, provided the correct use of the methodology.

Finally, after all developments, we can assure that Large Eddy Simulation, as well as other important methodologies like the Immersed Boundary Method are not only promising methodologies, but they can be looked as engineering tools in the design of new processes and equipments.

ACKNOWLEDGMENTS

RESUMO

O presente trabalho é dedicado ao desenvolvimento e implementação de uma estrutura computacional para realizar simulações numéricas de escoamentos turbulentos sobre geometrias complexas a baixo número de Mach. O algoritmo em análise é baseado em um esquema de integração no tempo preditor-corretor clássico, que utiliza um método de projeção para as equações de transporte de quantidade de movimento. A estratégia de decomposição de domínio é adotado para computação distribuída, exibindo níveis muito satisfatórios de aumento de velocidade e eficiência. A Metodologia da Fronteira Imersa é utilizada para caracterizar a presença de uma geometria complexa. Este método exige duas malhas computacionais separadas: Uma Euleriana, onde as equações de transporte são resolvidas com o método do Volume finito, de discretização de segunda ordem e um domínio Lagrangiano, representado por uma malha tipo casca não-estruturada que representa a geometria imersa. O código desenvolvido foi totalmente verificado pelo método das Soluções Manufaturadas, em ambos os domínios Euleriano e de Lagrangiano. As capacidades da estrutura computacional resultante são ilustradas em quatro casos distintos: um jato turbulento, o escoamento de Poiseuille, como uma questão de validação da Metodologia da Fronteira Imersa implementada, o escoamento sobre uma esfera,

que abrange uma ampla gama de números de Reynolds, e finalmente, com a intenção de demonstrar a aplicabilidade da metodologia das Simulações das Grandes Escalas - LES - em um problema industrial, o escoamento turbulento dentro de um ventilador industrial.

Palavras-chave: Escoamentos Turbulentos, Simulação de Grandes Escalas, Método da Fronteira Imersa, Escoamentos a Baixo Número de Mach, Escoamentos Industriais.

REFERENCES

CAMPEGHER R. 2005. Análise Numérica do Processo de Transferência de Calor sobre Corpos Vibrantes, Aplicação do Método de Fronteira Imersa com Modelo Físico Virtual. Ph.D. thesis. Universidade Federal de Uberlândia.

CAMPREGHER R. 2005. Modelagem matemática tridimensional para problemas de interção fluido-estrutura. Ph.D. thesis. Universidade Federal de Uberlândia. Uberlândia.

CHEN Y, PETERS N, SCHNEEMANN G, WRUCK N, RENZ U AND MANSOUR M. 1996. The detailed flame structure of highly stretched turbulent premixed methane-air flames. Combust Flame 107: 223-226.

DA-SILVA CFNB. 2001. The role of coherent structures in the control and interscale interactions of round, plane and coaxial jets. Ph.D. thesis. Institute National Polytechnique de Grenoble. Grenoble.

DANAILA I AND BOERSMA BJ. 2000. Direct numerical simulation of bifurcating jets. Phys Fluids 12: 1255-1257.

FERZIGER J AND PERIC M. 1996. Computational methods for fluid dynamics. Springer.

FORNBERG B. 1988. Steady viscous flow past a sphere at high reynolds number. J Fluid Mech 190: 471-489.

GERMANO M, PIOMELLI U, MOIN P AND CABOT WH. 1991. A dynamic subgrid-scale eddy viscosity model. Physics of Fluids A: Fluid Dynamics 3: 1760.

GRIFFITH BE AND PESKIN CS. 2005. On the order of accuracy of the immersed boundary method: Higher order convergence rates for sufficiently smooth problems. J Comput Phys 208: 75-105.

HAWORTH D. 2010. Progress in probability density function methods for turbulent reacting flows. Prog Energ Combust 36: 168-259.

JEONG J AND HUSSAIN F. 1995. On the identification of a vortex. J Fluid Mech 285: 69-94.

LESIEUR M AND MÉTAIS O. 1996. New Trends in Large-Eddy Simulations of Turbulence. Annu Rev Fluid Mech 28: 45-82.

LESIEUR M, MÉTAIS O AND COMTE P. 2005. Large-Eddy Simulations of Turbulence. Cambridge University Press, New York.

LESSANI B AND PAPALEXANDRIS M. 2006. Time-accurate calculation of variable density flows with strong temperature gradients and combustion. J Comput Phys 212: 218-246.

LILLY D. 1992. A proposed modification of the Germano subgrid-scale closure method. Physics of Fluids A: Fluid Dynamics 4: 633.

MARIANO FP, MOREIRA LQ, SILVEIRA-NETO A, DA SILVA CB AND PEREIRA JCF. 2010. A new incompressible navier-stokes solver combining fourier pseudo-spectral and immersed boundary method. CMES - Comp Model Eng 59: 181-216.

MENEVEAU C, LUND TS AND CABOT WH. 2006. A Lagrangian dynamic subgrid-scale model of turbulence. J Fluid Mech 319: 353-385. Published online: Apr 2006 doi: 10.1017/S0022112096007379

MOIN P, SQUIRES K, CABOT W AND LEE S. 1991. A dynamic subgrid? scale model for compressible turbulence and scalar transport 3: 2746-2757.

PIERCE CD. 2001. Progress-Variable Approach For Large-Eddy Simulation of Turbulent Combustion. Doctor of philosophy. Stanford University.

POINSOT T AND VEYNANTE D. 2005. Theoretical and Numerical Combustion, Second Edition. R.T. Edwards, Inc.

POPE SB. 2000. Turbulent Flows. Cambridge University Press, p. 771.

SCHNEIDER GE AND ZEDAN M. 1981. A modified strongly implicit procedure for the numerical solution of field problems. Numer Heat Transfer 4: 1-19.

SCOTTI A, MENEVEAU C AND LILLY DK. 1993. Generalized Smagorinsky model for anisotropic grids. Physics of Fluids A: Fluid Dynamics 5: 2306.

SILVEIRA-NETO A, GRAND D, MÉTAIS O, GONZE M AND LESIEUR M. 1993. A numerical investigation of the coherent vortices in turbulence behind a backward-facing step. J Fluid Mech 256: 1-25.

SMAGORINSKY J. 1963. General circulation experiments with the primitive equations. Mon Weather Rev 91: 99-164.

SMIRNOV A. 2004. Random flow generation procedure technical manual, p. 1-13.

SMIRNOV A, SHI S AND CELIK I. 2001. Random flow generation technique for large eddy simulations and particle-dynamics modeling. J Fluids Eng 123: 359-371.

STEINBERG S AND ROACHE PJ. 1985. Symbolic manipulation and computational fluid dynamics. J Comput Phys 57: 251-284.

SUBRAMANIAN RS. 2003. Drag on a sphere. Notas de aula do curso de Mecânica dos Fluidos no Departamento de Engenharia Química da Clarkson University, New York .

UHLMANN M. 2005. An immersed boundary method with direct forcing for the simulation of particulate flows. J Comput Phys 209: 448-476.

VAN DER VORST HA. 1981. Iterative solution methods for certain sparse linear systems with a non-symmetric matrix arising from pde-problems. J Comput Phys 44: 1-19.

VEDOVOTO JM. 2007. Modelagem matemática de simulação numérica de escoamentos incompressíveis sobre geometrias complexas tridimensionais utilizando o método da fronteira imersa. Master's thesis. Universidade Federal de Uberlândia. Uberlândia. (Unpublished).

VEDOVOTO JM. 2011. Mathematical and numerical modeling of turbulent reactive flows using a hybrid LES / PDF methodology. Ph.D. thesis. Universidade Federal de Uberlândia and École Nationale Supérieure de Mécanique et d'Aérotechnique. Uberlândia.

VEDOVOTO JM, DA SILVEIRA NETO A, MURA A AND FIGUEIRA DA SILVA LF. 2011. Application of the method of manufactured solutions to the verification of a pressure-based finite-volume numerical scheme. Computers & Fluids 51: 85-99.

WANG Z, FAN J AND LUO K. 2008. Combined multi direct forcing immersed boundary method for simulating flows with moving particles. Int J Multiphas Flow 34(3): 283-302.

WHITE FM. 1999. Mecânica dos Fluidos. McGraw Hill, 4ª ed., 570 p.

YILMAZ S. 2008. RANS/PDF and LES/FDF for prediction of turbulent premixed flames. Ph.D. thesis. University of Pittsburg.

YILMAZ S, NIK M, SHEIKHI M, STRAKEY P AND GIVI P. 2011. An irregularly portioned lagrangian monte carlo method for turbulent flow simulation. J Sci Comput 47: 109-125.

YOU D AND MOIN P. 2008. A dynamic global-coefficient subgrid-scale model for compressible turbulence in complex geometries. Center for Turbulence Research, p. 189-196.