A 2D Software System for Expedite Analysis of CFD Problems in Complex Geometries

A. M. G. LOPES

ADAI – *Department of Mechanical Engineering, University of Coimbra*, 3030 *Coimbra, Portugal*

Received 27 March 2015; accepted 18 May 2015

ABSTRACT: EasyCFD_G, a 2D software tool for fluid dynamics and heat transfer applications, is presented. The underlying physical and numerical principles are described, including mesh generation. The software is based on quadrilateral structured and unstructured meshes, rendering it with great versatility in geometrical terms. Amenable problems include multi-domain simulations with fluid and solid regions, thermal effects and mixing of different fluids, in stationary or turbulent regime. For illustration, a few example problems are presented, covering different physical situations, including comparison with benchmark data. Due to its versatility, fast learning curve and easy and intuitive utilization, this software tool represents a great value for educational purposes. Comput Appl Eng Educ 24:27–38, 2016; View this article online at wileyonlinelibrary.com/journal/cae; DOI 10.1002/cae.21668

Keywords: computational fluid dynamics; heat transfer; numerical methods; mesh generation; introduction

INTRODUCTION

Nowadays, several software packages are available for the simulation of fluid flow problems, including heat transfer and other physical phenomena such as multiphase flow, reacting fluids, etc. Some of these tools are commercialized at a cost that often is either not accessible to enterprises and universities or are not cost effective for sparse utilization. As alternative, some free codes, or free versions of commercial codes, are also available, some of them with a rather easy usage, but others lacking user-friendliness and with a long learning curve. The software presented in this paper, EasyCFD_G, was designed for the numerical solution of fluid flow and heat transfer in two-dimensional geometries. The main guiding lines on its development were the simplicity and the intuitiveness in its utilization, in a self-contained package. It represents a major evolution of the older software EasyCFD, described in Ref. [1], which was limited to simple geometries delimited by straight horizontal and vertical boundaries. The utilization of quadrilateral unstructured meshes allows the simulation to deal with complex geometries of virtually any configuration. Support is also provided for axisymmetric situations. EasyCFD_G is able to deal with stationary and transient regime, laminar and turbulent flow (*k*-ε and *SST k*-ω turbulence models are available), including thermal effects and heat conduction in solids, dispersion of passive scalars, and two-component single phase fluid flow. The software is supplied as an executable, with no access to source code. Due to the simple and intuitive user interface, this software tool constitutes a valuable tool for educational purposes, although it can also be applied for practical cases, within the limitations of a 2D approach.

Correspondence to A. M. G. Lopes (antonio.gameiro@dem.uc.pt).

THEORETICAL BACKGROUND

In the following, a brief explanation of the theoretical principles of computational fluid dynamics behind EasyCFD_G is given. Details on physical and numerical models may be found in the cited references and also on the software usেr's manual (www.easycfd.net).

Basic Transport Equations

For the description of the transport equations, the x and z coordinates of the Cartesian coordinate system shall be taken as the independent variables, to which correspond, for velocity, the u and the w components. The coordinate transformation for the generalized mesh will be referred later on the present work.

The Navier–Stokes equations describe momentum conservation and, for a 2D situation, may be stated as follows:

$$\frac{\partial(\rho u)}{\partial t}+\frac{\partial}{\partial x}\left(\rho u^2\right)+\frac{\partial}{\partial z}(\rho u w) =\frac{\partial}{\partial x}\left[\Gamma\left(2\frac{\partial u}{\partial x}-\frac{2}{3}div\vec{V}\right)\right]+\frac{\partial}{\partial z}\left[\Gamma\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right]-\frac{\partial p}{\partial x} \quad (1)$$

$$\frac{\partial(\rho w)}{\partial t}+\frac{\partial}{\partial x}(\rho u w)+\frac{\partial}{\partial z}\left(\rho w^2\right) =\frac{\partial}{\partial z}\left[\Gamma\left(2\frac{\partial w}{\partial z}-\frac{2}{3}div\vec{V}\right)\right]+\frac{\partial}{\partial x}\left[\Gamma\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)\right] -\frac{\partial p}{\partial z}+I \quad (1a)$$

where p [N/m^2] stands for pressure, $\vec{V}$ is the velocity vector, ρ [kg/m^3] is density and I represents the buoyancy forces. The diffusion coefficient includes the contributions of the dynamic and turbulent viscosity, i.e., $\Gamma=\mu+\mu_t$ [$N\ s/m^2$].

In turn, the conservation of mass law, or continuity equation, is:

$$\frac{\partial \rho}{\partial t}+\frac{\partial}{\partial x}(\rho u)+\frac{\partial}{\partial z}(\rho w)=0 \quad (2)$$

The energy conservation equation is obtained considering the transport equation for the enthalpy $\varphi=c_p T$, where c_p [$J/kg\ K$] is the fluid specific heat and T [K] is its temperature. For a fluid medium, the corresponding equation is:

$$\frac{\partial\left(\rho c_p T\right)}{\partial t}+\frac{\partial}{\partial x}\left(\rho c_p u T\right)+\frac{\partial}{\partial z}\left(\rho c_p w T\right) =\frac{\partial}{\partial x}\left(\Gamma\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial z}\left(\Gamma\frac{\partial T}{\partial z}\right)+S_T \quad (3)$$

with the source term S_T representing the heat generation rate per unit volume [W/m^3] and $\Gamma=(\mu/\mathrm{Pr}+\mu_t/\mathrm{Pr}_t)c_p$, with Pr and Pr_t as the laminar and the turbulent Prandtl number. For solid regions, heat transfer is governed by conduction:

$$\frac{\partial\left(\rho c_p T\right)}{\partial t}=\frac{\partial}{\partial x}\left(\lambda\frac{\partial T}{\partial x}\right)+\frac{\partial}{\partial z}\left(\lambda\frac{\partial T}{\partial z}\right)+S_T \quad (4)$$

where λ [W/mK] is the thermal conductivity of the material. In a multi-component fluid flow situation, different fluids are present, sharing the same velocity, pressure, temperature, and turbulence quantities. EasyCFD_G contemplates the case of two fluids, with a single phase. The concentration of one of the components (the secondary fluid) is computed with a normal transport equation for a scalar:

$$\frac{\partial(\rho\phi_{c2})}{\partial t}+\frac{\partial}{\partial x}(\rho u\phi_{c2})+\frac{\partial}{\partial z}(\rho w\phi_{c2}) =\frac{\partial}{\partial x}\left(\Gamma\frac{\partial\phi_{c2}}{\partial x}\right)+\frac{\partial}{\partial z}\left(\Gamma\frac{\partial\phi_{c2}}{\partial z}\right) \quad (5)$$

where φ_{c2} is the mass fraction or concentration of the secondary fluid. The diffusion coefficient is given by:

$$\Gamma=\rho D_{\phi_c}+\frac{\mu_t}{Sc_t} \quad (6)$$

where D_{φ_c} [m^2/s] is the kinematic diffusivity that characterizes the fluids pair and Sc_t is the turbulent Schmidt number. The concentration of the secondary fluid plays an active role in the flow field, since fluid properties depend on the concentration of each component. The mixture properties (such as viscosity, specific heat, etc.) are determined by a weighted average of the properties of the components.

Turbulence Modelling

Two of the most popular turbulence models are available, as described next.

The k-ε Turbulence Model. The standard formulation of this turbulence model is described in [2–4]. The turbulent viscosity is given by:

$$\mu_t=C_\mu\frac{\rho k^2}{\varepsilon} \quad (7)$$

The turbulence kinetic energy, k [m^2/s^2], and its dissipation rate, ε [m^2/s^3], are computed with the following transport equations:

$$\frac{\partial(\rho k)}{\partial t}+\frac{\partial}{\partial x}(\rho u k)+\frac{\partial}{\partial z}(\rho w k)=\frac{\partial}{\partial x}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial x}\right] +\frac{\partial}{\partial z}\left[\left(\mu+\frac{\mu_t}{\sigma_k}\right)\frac{\partial k}{\partial z}\right]+P_k-\rho\varepsilon+G_T \quad (8)$$

$$\frac{\partial(\rho\varepsilon)}{\partial t}+\frac{\partial}{\partial x}(\rho u\varepsilon)+\frac{\partial}{\partial z}(\rho w\varepsilon)=\frac{\partial}{\partial x}\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial\varepsilon}{\partial x}\right] +\frac{\partial}{\partial z}\left[\left(\mu+\frac{\mu_t}{\sigma_\varepsilon}\right)\frac{\partial\varepsilon}{\partial z}\right]+\frac{\varepsilon}{k}(C_1P_k-C_2\rho\varepsilon+C_3G_T) \quad (9)$$

The term P_I represents the production rate of k as the result of velocity gradients:

$$P_k=\mu_t\left[2\left(\frac{\partial u}{\partial x}\right)^2+2\left(\frac{\partial w}{\partial z}\right)^2+\left(\frac{\partial u}{\partial z}+\frac{\partial w}{\partial x}\right)^2\right] \quad (10)$$

while the term G_T accounts for the production or destruction of k and ε due to the thermal gradients:

$$G_T=-\beta g\frac{\mu_t}{\mathrm{Pr}_t}\frac{\partial T}{\partial z} \quad (11)$$

with g [m/s^2] as the gravity acceleration and β [K^{-1}] as the fluid dilatation coefficient. The remaining model constants are:

$$C_\mu=0.09;\quad \sigma_k=1.0;\quad \sigma_\varepsilon=1.3; \quad C_1=1.44;\quad C_2=1.92;\quad C_3=1.44 \quad (12)$$

The SST Turbulence Model. The *SST* model [5] represents a combination of the *k*-ε and the *k*-ω models. According to Ref. [6], the *k*-ω model is more accurate near the wall but presents a high sensitivity to the ω values in the free stream region, where the *k*-ε model shows a better behavior. The *SST* model represents a blend of the two, through a weighting factor computed based on the nearest wall distance. The governing equations are:

$$\frac{\partial(\rho k)}{\partial t}+\frac{\partial(\rho u k)}{\partial x}+\frac{\partial(\rho w k)}{\partial z} =\frac{\partial}{\partial x}\left((\mu+\sigma_k\mu_t)\frac{\partial k}{\partial x}\right)+\frac{\partial}{\partial z}\left((\mu+\sigma_k\mu_t)\frac{\partial k}{\partial z}\right)+\bar{P}_k-\beta\,\rho\omega\mathrm{k} \quad (13)$$

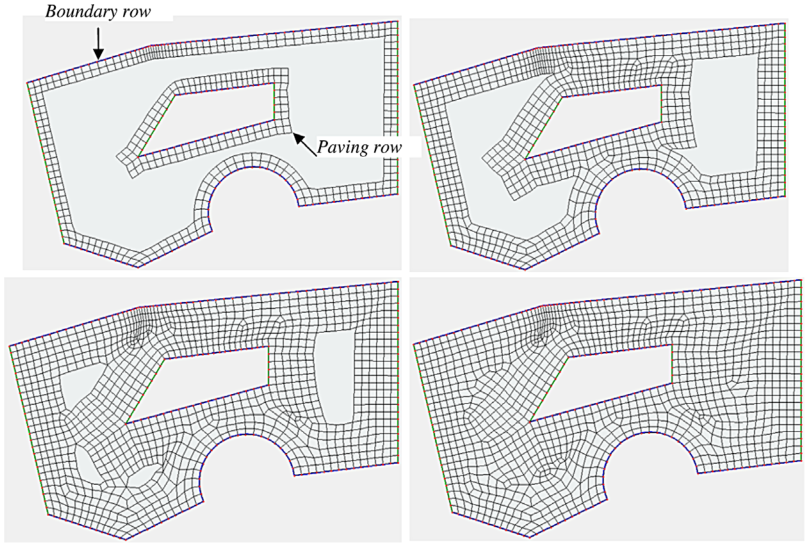


Figure 1 Paving process for unstructured mesh generation. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

$$\frac{\partial(\rho\omega)}{\partial t}+\frac{\partial(\rho u\omega)}{\partial x}+\frac{\partial(\rho w\omega)}{\partial z}=\frac{\partial}{\partial x}\left((\mu+\sigma_\omega\mu_t)\frac{\partial\omega}{\partial x}\right)+\frac{\partial}{\partial z}\left((\mu+\sigma_\omega\mu_t)\frac{\partial\omega}{\partial z}\right)+\frac{\alpha\bar{P}_k}{\nu_t}-\beta\rho\omega^2+2(1-F_1)\rho\sigma_{\omega 2}\frac{1}{\omega}\left(\frac{\partial k}{\partial x}\frac{\partial\omega}{\partial x}+\frac{\partial k}{\partial z}\frac{\partial\omega}{\partial z}\right) \tag{14}$$

where ω is the frequency of dissipation of turbulent kinetic energy $[s^{-1}]$. The production of turbulent kinetic energy is limited to prevent the build-up of turbulence in stagnant regions:

$$\bar{P}_k=\min(P_k,10\beta*\rho k\omega) \tag{15}$$

The weighting function F_1 is given by:

$$F_1=\tanh\left\{\left\{\min\left[\max\left(\frac{\sqrt{k}}{\beta*\omega y};\frac{500\nu}{y^2\omega}\right);\frac{4\rho\sigma_{\omega 2}k}{CD_{k\omega}y^2}\right]\right\}^4\right\} \tag{16}$$

and

$$CD_{k\omega}=\max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial\omega}{\partial x_j},10^{-10}\right) \tag{17}$$

where y represents the distance to the neighbour wall and ν is the laminar dynamic viscosity. F_1 is zero away from the wall (k-ε model) and changes to unit inside the boundary layer (k-ω model), with a smooth transition based on y. The turbulent viscosity is computed as:

$$\nu_t=\frac{a_1k}{\max(a_1\omega;SF_2)} \tag{18}$$

where S is the invariant measure of the strain rate. The function F_2 has the following formulation:

$$F_2=\tanh\left\{\left[\max\left(\frac{2\sqrt{k}}{\beta*\omega y};\frac{500\nu}{y^2\omega}\right)\right]^2\right\} \tag{19}$$

The values for σ_k, σ_ω, α, and β are computed as a blend of the corresponding values for the k-ε and the k-ω models, through the following generic equation:

$$\phi=F_1\phi_1+(1-F_1)\phi_2 \tag{20}$$

The constants are $\alpha_1=5/9$; $\beta_1=3/40$; $\sigma_{k1}=0.85$; $\sigma_{\omega 1}=0.5$; $\alpha_2=0.44$; $\beta_2=0.0828$; $\sigma_{k2}=1$; $\sigma_{\omega 2}=0.856$; $\beta^*=0.09$; a_1=0.31.

Numerical Method

Integration and Solution of the Equations. Discretization and integration of the transport equations described previously are performed using a non-orthogonal generalized mesh composed of quadrilateral elements. The independent Cartesian coordinates (x, z), describing the physical domain, are, thus, replaced by a boundary fitted coordinate system (ξ, ζ), defined by the mesh lines

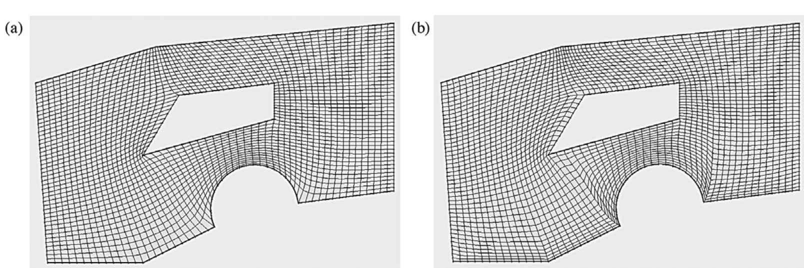


Figure 2 Structured mesh generation. (a) Laplacian generation; (b) Poisson generation.

which may have, locally, any orientation and inclination. The mesh arrangement is of the collocated type (as opposed to the staggered mesh), with the two velocity components and scalar quantities (temperature, pressure, turbulence kinetic energy and its dissipation rate, as well as concentrations) located at each control volume center.

The integration and solution method for the transport equations are entirely based on the control-volume described in Ref. [6]. The coupling between momentum and continuity equations is performed through the *SIMPLEC* algorithm (Semi-Implicit Method for Pressure-Linked Equations—Consistent) [7], which is based on the original formulation *SIMPLE* [6]. Due to the non-staggered mesh arrangement (collocated mesh), the Rie–Chow interpolation procedure [8], with the modifications proposed in Refs. [9,10], is implemented. Evaluation of variables (velocity components, temperature, etc.) at the control volume faces is performed using a wide choice of advection methods (upwind, Quick, and several TVD schemes are available, as described in Refs. [6,11]). After integration, the original differential equations are cast in the standard formulation, as follows:

$$a_P\phi_P = a_E\phi_E + a_O\phi_O + a_T\phi_T + a_B\phi_B + b \qquad (21)$$

stating that the value of each variable φ ($\varphi = u, w, T, k, \epsilon$) at a certain location P is directly related to the neighbor values. The solution of the equations is carried out with an iterative procedure. The Point Gauss–Seidel method is used for the scalar fields, while the Point Successive Over Relaxation method [12], in conjunction with the additive correction multigrid method [13], is adopted for the pressure correction [6]. The whole flow field calculation is considered to be converged when all normalized residuals are lower than a predefined value.

Mesh Generation

It is well known that the flow solution quality is greatly dependent on the mesh characteristics. The major requirements for a good mesh are:

- Higher resolution in regions where the flow characteristics are expected present higher gradients, typically near solid boundaries, and near recirculation regions;
- Smoothness, in the sense that the mesh spacing should change gradually, to reduce discretization errors;
- Low degree of skewness.

Mesh generation is a procedure for the calculation of the Cartesian coordinates (x, z) of each mesh node. Two types of quadrilateral meshes are available within EasyCFD_G: structured meshes and unstructured meshes. Structured meshes allow more control on the final result, but are more complex to setup. Unstructured meshes are easier to setup, but the user has, to a certain extent, less control. Overall, unstructured meshes are the best option, due to their versatility. Both types of mesh generation

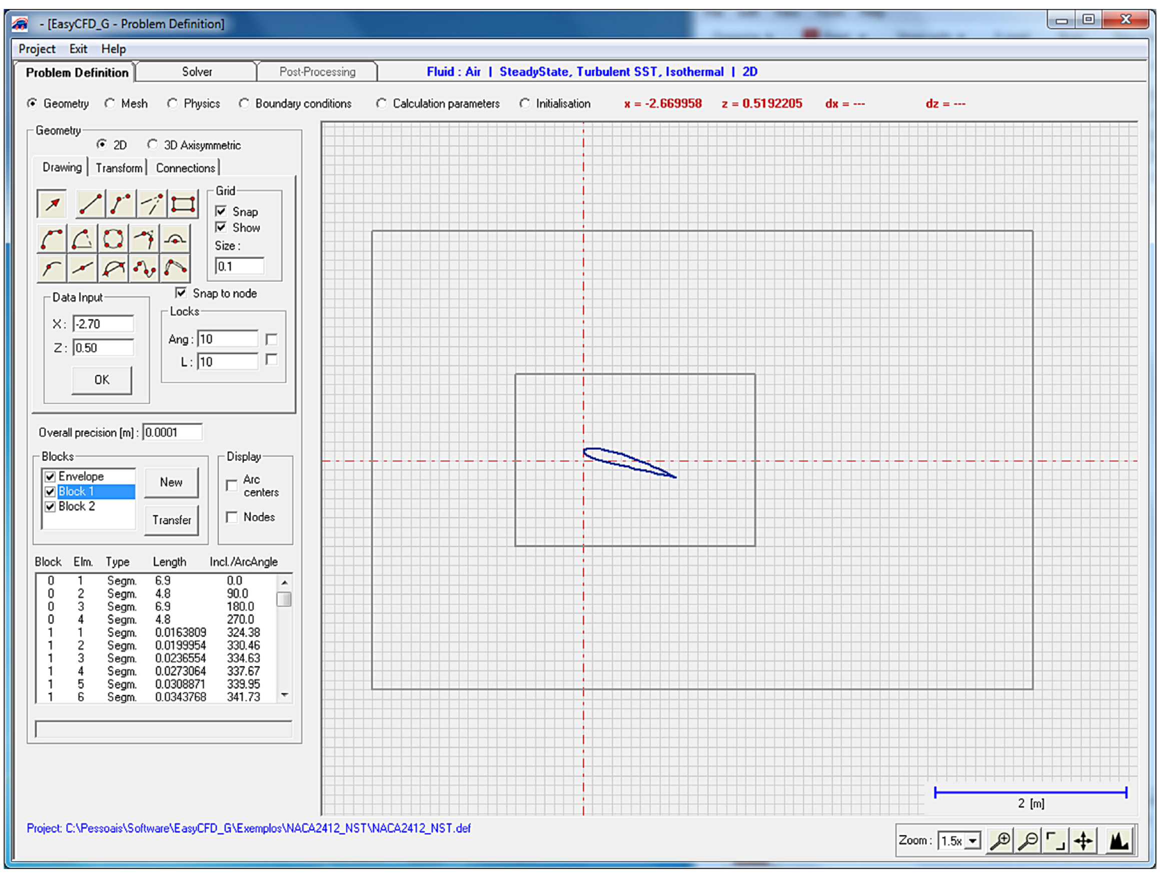


Figure 3 General view of the user interface for the Problem Definition—Geometry section. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

methods compute the location of inner nodes starting from the 1D domain boundary nodes distribution.

The unstructured mesh generation method (called Paving Method) implemented in EasyCFD_G closely follows the proposal described in detail in Ref. [14]. Basically, following this method, nodes coordinates are computed starting from the boundary nodes distribution. A paving row (row of nodes which is being computed) starts from a boundary row (row lying in a geometric boundary), by adding nodes to the previous row, as depicted in Figure 1. The process continues until the whole domain is filled. Mesh growth can be continuously visualized, allowing the user to identify potential problems and to take measures to correct them.

The numerical procedure implemented in EasyCFD_G for structured mesh generation is based, in a considerable extent, on the method described in Refs. [15,16]. Numerical mesh generation using *Laplace* differential equations (and *Poisson* differential equations) is mostly used due to the associated smoothness of the meshes produced by these methods. The *Laplace* system has the following formulation:

$$\xi_{xx} + \xi_{zz} = 0 \quad ; \quad \zeta_{xx} + \zeta_{zz} = 0 \tag{22}$$

where the double index represents the second derivative. These equations state, in fact, that the x and z coordinates of a mesh node are obtained from the arithmetic average of the x and z coordinates of its neighbor nodes. Figure 2a shows a mesh generated with the Laplace system. Due to its averaging characteristics, the Laplace system tends to produce equally distributed nodes inside the domain, leading to a poor mesh resolution near boundaries. Control of mesh characteristics is possible by solving a Poisson equation system, instead of the Laplace equation system:

$$\xi_{xx} + \xi_{zz} = P \quad ; \quad \zeta_{xx} + \zeta_{zz} = Q \tag{23}$$

In this set of *Poisson* equations, P and Q are forcing terms allowing the control of the mesh lines clustering near the boundaries. As a result mesh spacing can be controlled, as exemplified in Figure 2b.

WORKING WITH EASYCFD_G

Setting up and solving a problem goes through three major stages to be accomplished sequentially: Problem Definition; Solver and Post-Processing. Each of these stages is briefly described next.

Problem Definition

The problem definition includes several steps:

Geometry—The problem geometry is constituted by an envelope enclosing the calculation domain. Inside the envelope, several closed blocks and open lines may be defined. Closed blocks can represent either void regions or meshed regions. Open lines may be used for mesh control or for representation of a thin

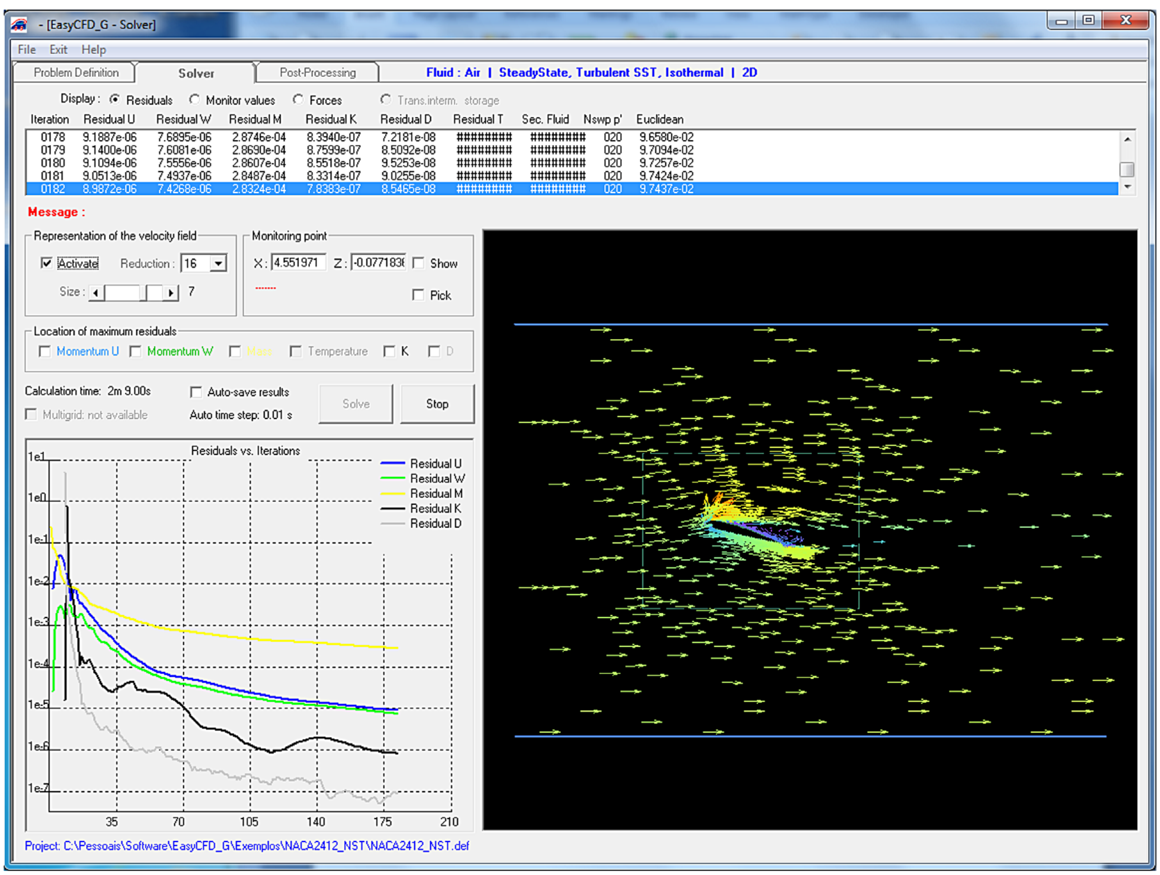


Figure 4 General view of the user interface for the Solver section. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

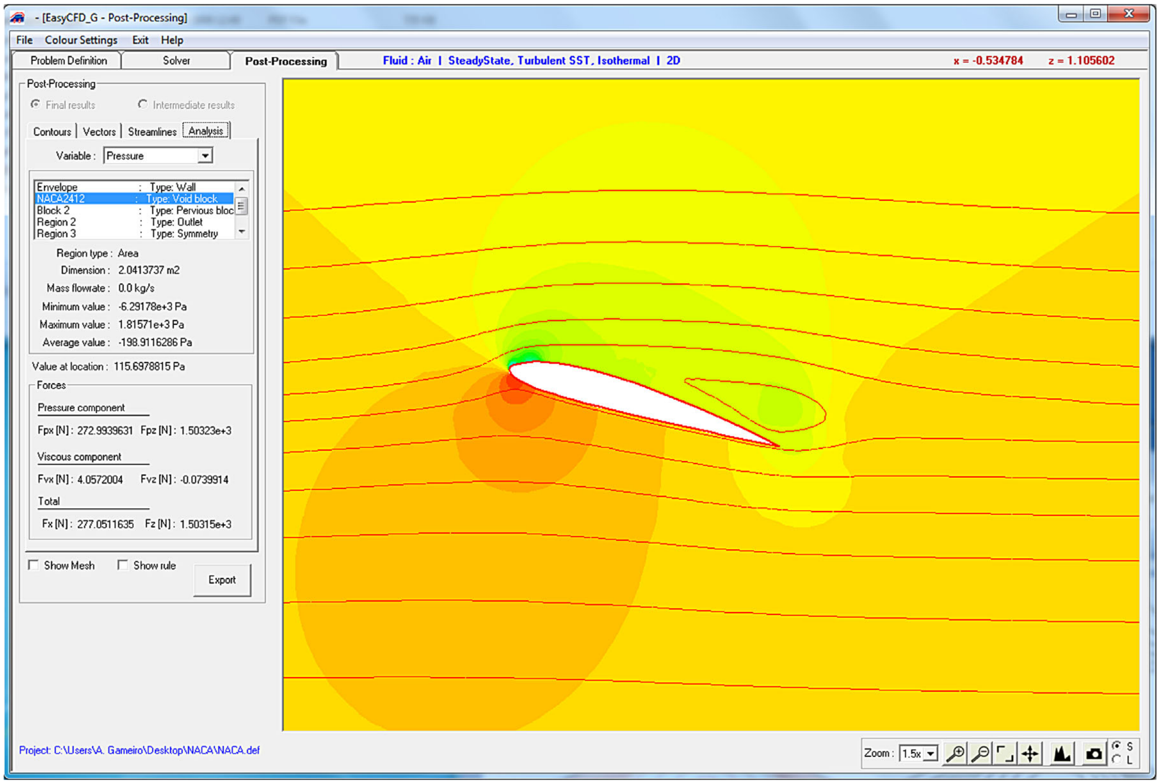


Figure 5 General view of the user interface for the Post-processing section. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

solid boundary. Several drawing tools are available, including splines. Worth mentioning is the Import feature, that allows reading DXF files or point coordinate ASCII files. Figure 3 shows the software interface, in the geometry drawing mode.

Mesh—For the mesh generation, several parameters may be defined, including boundary nodes distribution, mesh clustering near boundaries, and target control volumes size, among others.

Physics—the definition of the whole problem physics includes the fluid properties, the flow regime (stationary or transient), flow type (laminar or turbulent), inclusion of thermal effects, turbulence model, etc.

Boundary Conditions—available types are inlet, outlet, symmetry, and wall. Specification of heat generation rate for volumetric regions, either of fluid or solid type, is included in this section. Boundary condition values can be either fixed, or given as a "Table Function," taking as the dependent variable either a space coordinate or time.

Calculation Parameters—the specification of numeric parameters is made on this section, including subrelaxation type (either False Transient or E-factor [7]), number of iterations, advection schemes, time step, and time interval for storing transient results.

Initialization—values for the different variables can be specified independently for each volumetric region. Alternatively, a finalized calculation may be used as the starting point for a new calculation (example: taking steady state results for starting a transient calculation).

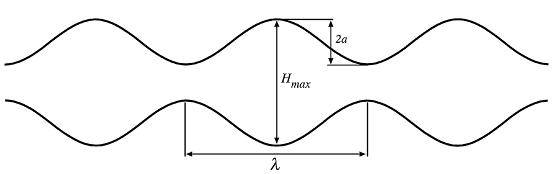


Figure 6 Geometry for the wavy channel.

Solver

The solution stage includes some useful features such as the continuous display of the velocity vectorial field and monitoring of

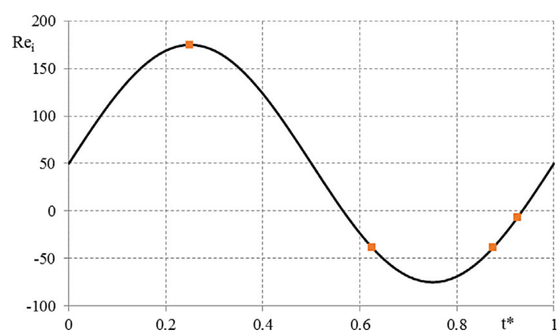


Figure 7 Instantaneous Reynolds number as function of the dimensionless time. Red dots represent time instants for Figure 10. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

the location of maximum residuals, so as to help identifying potential causes in the event of convergence problems. Monitoring of variables at selected locations as well as integral quantities such as forces in regions are also available. A sample image of graphical interface is given in Figure 4. Running times are quite modest. As an example, the flow around the airfoil presented in section 4.3 takes 12 min of running time in a modern Intel® Core(TN) i7-5930K @ 3.5GHz CPU, for a 120,000 nodes mesh. This is a quite refined mesh, used to obtain accurate lift and drag results. For first analysis and flow topology visualization, meshes of around 20,000 nodes may be used, leading to a run time of less than 2 min.

Post-Processing

For post-processing, several visualization tools are available, such as vector and streamlines representation for the velocity field and color contours for scalars. Quantitative analysis computes average and minimum and maximum values in selected regions. Data exporting is available either for ASCII text files or directly into Microsoft Excel sheets. For transient problems, it is possible to export data using the formats just described, for the whole transient process, allowing the generation of graphics for the transient evolution of variables. The software interface for the post-processing is depicted in Figure 5, showing the pressure color contours for the air flow around a NACA 2412 airfoil. The represented streamlines are selected by the user, by clicking locations on the visualization window.

APPLICATION CASES

In the following, some application cases are presented, illustrating different physical situations that are commonly addressed in class, such as transient flows, airfoil drag computation, mixing of fluids, and problems involving heat transfer. Whenever available, comparison with benchmark data is presented.

Pulsatile Flow Inside a Symmetric Channel With Wavy Walls

Increasing heat transfer rate in laminar flows is of great importance in areas such as biomedicine and biochemistry. As reported in Ref. [17], the combination of flow separation with a pulsed transient behavior may prove to be an excellent solution when the objective is to promote the transfer processes of heat and mass, with the applications such as blood oxygenators. In Ref. [17], this type of flow was studied through experimental and numerical approaches applied to a two-dimensional channel with sinusoidal walls. The choice of this application case aims to illustrate the capability of the software to deal with time-dependent boundary conditions, including boundaries that may be of inlet or outlet type, depending on the time instant. The geometrical configuration is presented in Figure 6.

The instantaneous inlet flow is characterized by:

$$Re_i(t) = Re_m(1 + A\sin(2\pi t/T)) \tag{24}$$

where t is time, T is the oscillation period, A is the oscillation amplitude and Re_m is the average Reynolds number:

$$Re_m = Q_m/\nu \tag{25}$$

with Q_m as the volumetric flowrate per unit depth. The temporal scale is given by the Womersley number, α^2:

$$\alpha^2 = H_{\max}^2 2\pi/(T\nu) \tag{26}$$

The dimensionless time t* is defined as $t^* = t/T$. For the present case, the following values where considered: $2a = 3.5mm$; $H_{\max} = 10mm$; $\lambda = 14mm$; $\alpha^2 = 650$; $A = 2.5$; $Re_m = 50$. Figure 7 presents the instantaneous Reynolds number as function of the dimensionless time, for an entire cycle.

In the present implementation, 3 temporal waves were considered in order to ensure a fully developed flow in temporal terms. Time-dependent velocity values are imposed at the two "inlets-outlets," using the feature "Table Function," as illustrated in Figure 8. Values at these boundaries have opposite signs, so that, at a certain time instant, there will always be one inlet and one outlet. Although two entire temporal cycles were simulated (in order to achieve a developed flow in temporal terms), only one cycled needs to be specified in the "Table Function" (since the first and the last velocity values are similar, periodic conditions are automatically assumed by the software). For the geometry, three waves were considered.

Visualizations are presented, in Figure 9, for streamlines at the central wave at the time instants identified by red dots in Figure 8. Comparison is made with corresponding results published in Ref. [17], which simulated this case using a vorticity-stream function formulation, in a triangular mesh. The flow presents a quite complex an interesting behavior, with vortices forming, detaching and disappearing along the time cycle. In spite of the fact that the streamlines values are not exactly the same in the two simulations (the values adopted for the streamlines in Ref. [17] are not reported), the obtained flow patterns and vortices size and location are essentially similar in both simulations. This is a quite interesting case for educational purposes, since it may be used to illustrate that the flow pattern in transient regimen may be quite different from any steady state solution at the same inlet velocity.

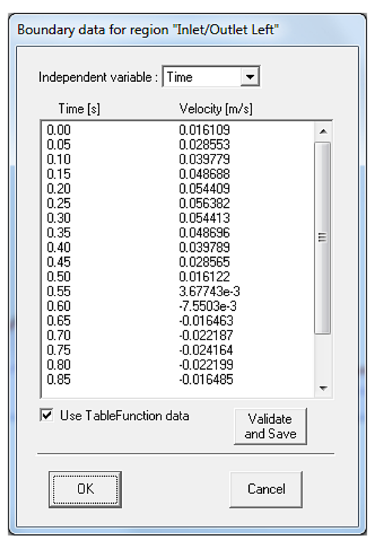


Figure 8 Table function for the imposition of time-dependent velocity values. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

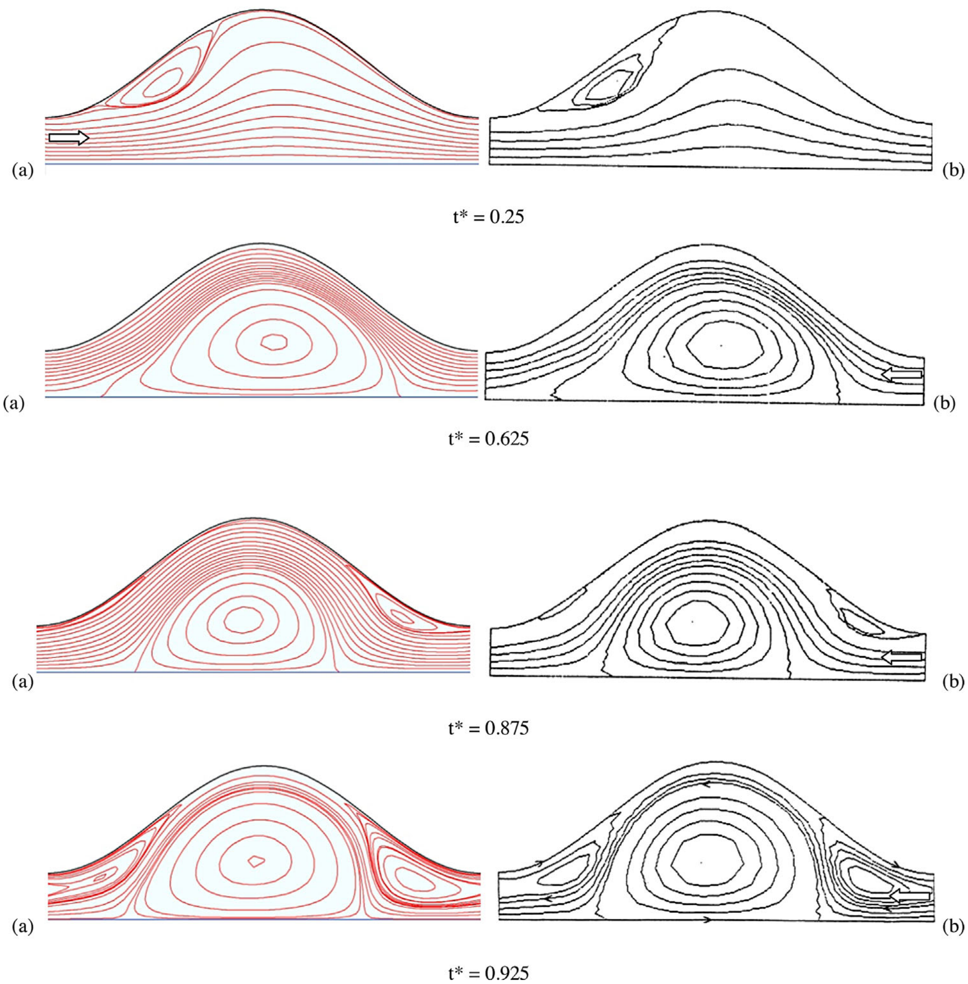


Figure 9 Streamlines visualization for different values for the adimensional time. (a): EasyCFD_G; (b): Nishimura e Matsune (1998). [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

Fully Developed Turbulent Flow Between Two Parallel Plates

For this test case, a *2D,* 1.2 *m* high channel is considered, for an air flow with an average velocity $\overline{V} = 1\, m/s$. Taking the air properties $\mu = 1.824 \times 10^{-5} N\,s/m^2$ and $\rho = 1.1884\, kg/m^3$, the flow Reynolds number is 39,092. As for the previous case, computations where performed in half the domain, with recourse to a symmetry boundary condition. In order to resolve the flow near the wall, a clustering distance of $1.5 \times 10^{-4}\, m$ was adopted on a structured mesh (cf. Figure 10). To obtain a developed velocity profile without using a very large length, the option "Profile," available in the software for structured meshes, was activated when imposing the inlet boundary conditions, allowing the transposition of a downstream profile into upstream, avoiding the utilization of a long calculation domain to produce developed conditions.

Computations where carried out using the *SST k-ω* turbulence model. Due to its hybrid characteristics, unlike the standard *k-ε* model, this model allows computations to be done in the viscous sublayer. The obtained results were compared with the direct numerical simulation presented [18], which were performed for a Reynolds number of 3300. Figure 11 displays the boundary layer profile obtained with both simulations. Note that, due to the lower Reynolds number adopted in Ref. [18], their range of u^+ values is not as large as the present one. One may note a very good agreement, except for the transition between the viscous and the log layer.

This problem is very interesting to show students how boundary layer flows presents similar solutions independently of

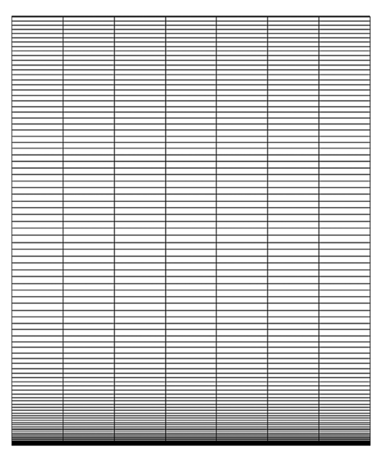


Figure 10 Mesh adopted for the fully developed channel flow.

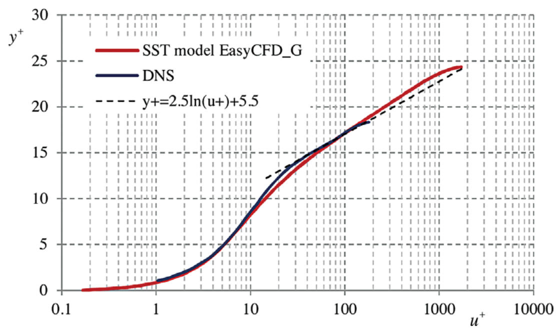


Figure 11 Boundary layer profile. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

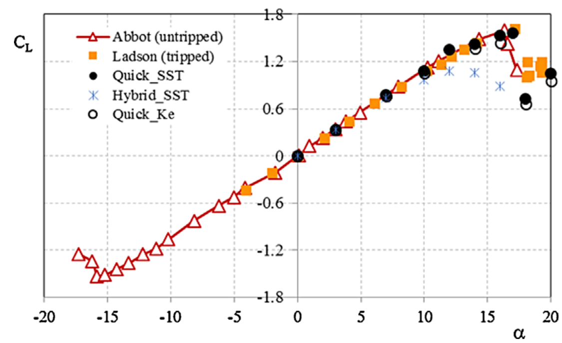


Figure 13 Calculation domain and mesh for NACA 0012 airfoil. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

the Reynolds number. A similar setup with an extended domain can also be used to show students the flow development from channel entrance up to the region of fully developed flow. Just by checking the axisymmetric option in the software interface, students may run the case for a pipe, in order to establish the comparison between 2D channel flow and pipe flow.

Drag and Lift Coefficient for the NACA 0012 Airfoil

The NACA 0012 airfoil encounters a wide range of applications, not only in aeronautics, where it is used mainly in airplanes and helicopters, but also in horizontal- and vertical-axis wind turbines. The numerical prediction of drag coefficients is not an easy task, as it is highly dependent on the correct simulation of the flow in the very close proximity of the wall. Specially demanding is the simulation of situations in the post-stall region, a common situation for wind turbines operation, especially of the vertical-axis type. In order to minimize boundary effects, a 80C long by 60C high domain was employed, where C is the airfoil chord ($=0.33$ m, for the simulated conditions). Present simulations were performed for a Reynolds number 6×10^6, which corresponds to a fully turbulent boundary layer. For the inlet boundary, a 5% turbulence intensity was assigned. After mesh independence tests, a total of approximately 130,000 control volumes were employed, as shown in Figure 12. A close view of the region around the airfoil is provided, evidencing the mesh quality near the airfoil surface.

Figure 13 shows the dependence of the lift coefficient with the airfoil angle of attack α. Both the SST and the k-ε turbulence models were used in these simulations. Experimental data is reported in Refs. [19,20] for the same Reynolds number. It is apparent that CFD results produced by both turbulence models are quite similar and agree very well with experimental data up to separation, which presents its onset at $\alpha = 16°$. Separation is completely established at $\alpha = 18°$ and for $\alpha \geq 20°$ the flow becomes unsteady, showing periodic oscillations on both aerodynamic coefficients, as may be appreciated in Figure 14 for a simulation in transient regime. In this figure, the initial instants up to $t = 0.2$ s correspond to the transition from the initial conditions up to the temporally developed state. As depicted in Figure 13, data in the stall region are quite spread, resulting also

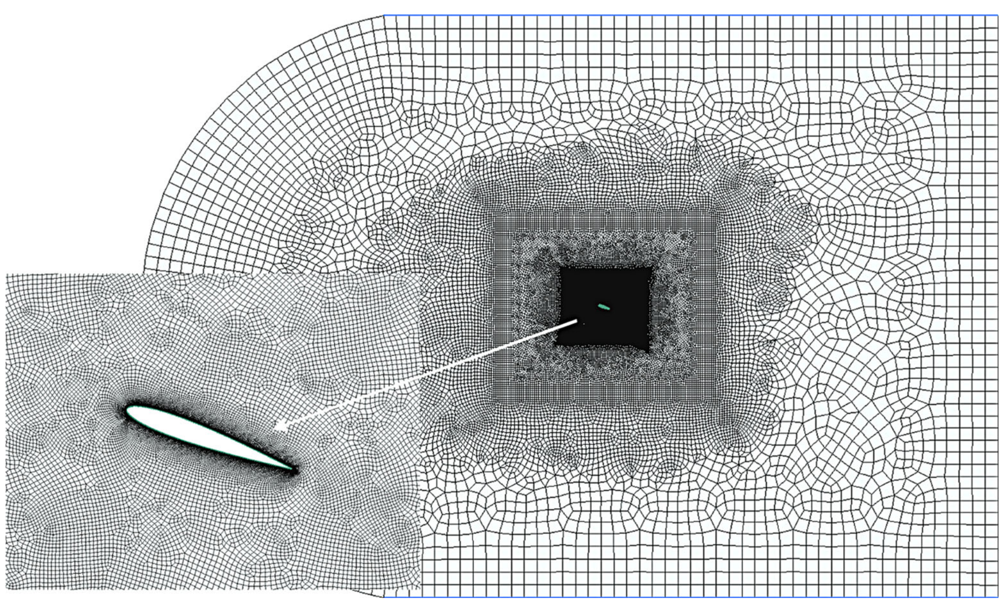


Figure 12 Calculation domain and mesh for NACA 0012 airfoil. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

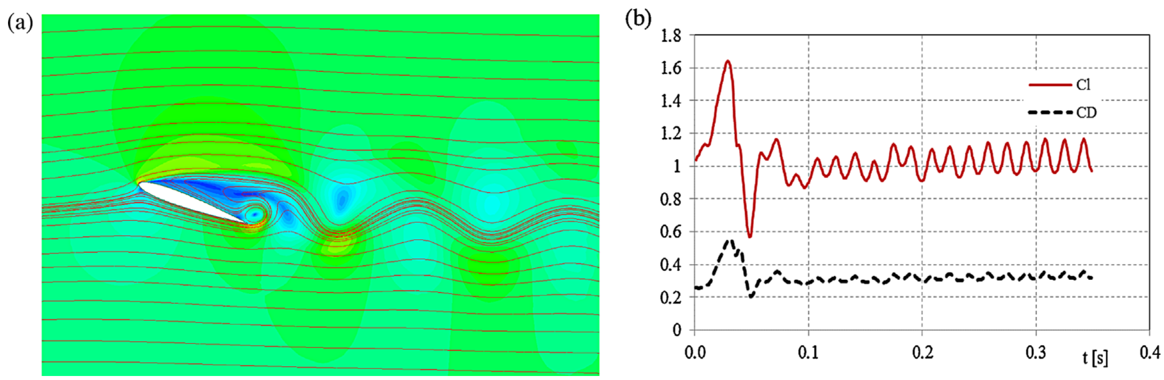


Figure 14 Stall at $\alpha = 20°$. (a) flow visualization at a frozen time instant; (b) time evolution of C_D. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

from the difficulty acknowledged by the authors on getting reliable experimental data, including ensuring effective 2D conditions. The relation between the drag and the lift coefficient is shown in Figure 15. It is interesting to note that deviations from experimental data are substantially larger for the *k-ε* model, certainly because, contrary to lift, where pressure forces dominate, the friction component plays an important role in drag and thus, correctly resolving the boundary layer becomes crucial. This case is very illustrative to students, as it shows that, even with steady boundary conditions, the flow solution may be unsteady. Is also exemplifies how apparently similar problems may present quite different degrees of agreement between experimental and numerical solutions (in this case, depending on the angle of attack).

Mixing of Different Fluids in a Pipe

This example illustrates different processes of mixing of two fluids inside a pipe. A flow of CO_2 is injected, through a pipe with diameter 20 cm, into a circular duct with diameter 1.2 m, where a flow of air exists. The initial velocity difference between both fluids is 0.6 m/s. Figure 16a shows the color contours for CO_2 concentration on the duct symmetry plane, for laminar diffusion. It is interesting to compare the solution when considering effects of turbulence on the flow. Turbulence enhances diffusion due to velocity fluctuations, resulting in a faster mixing between fluids, as shown in Figure 16b. Fluid mixing can be further enhanced, by placing obstacles within the flow. The resulting velocity field, included in the visualization of Figure 16c, is characterized by a series of vortices which, with the associated increased levels of turbulence, further enhances mixing. CO_2 concentration at the exit plane and at the duct centerline are shown in Figures 17 and 18, respectively, for the three considered cases.

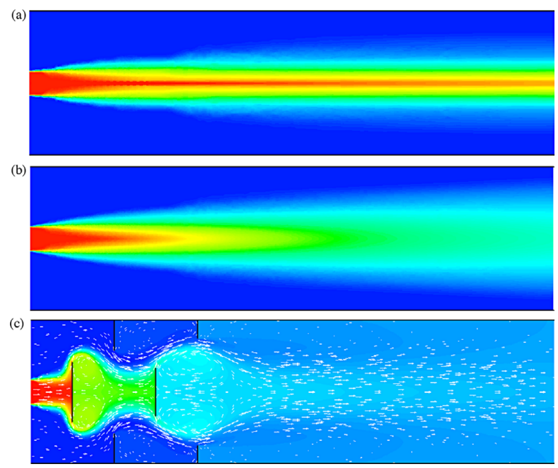


Figure 16 Distribution of CO_2 concentration for a pipe flow. (a) Laminar flow; (b) Turbulent flow; (c) Turbulent flow with baffles. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

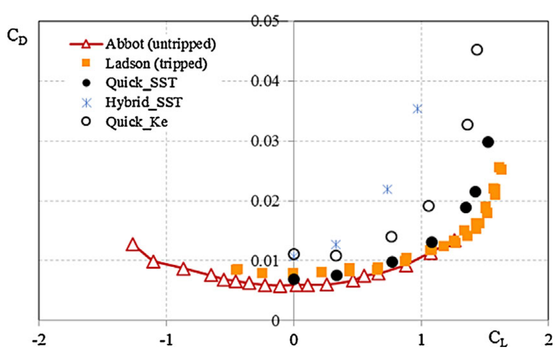


Figure 15 Drag coefficient *vs* lift coefficient. NACA 0012 airfoil. $Re = 6 \times 10^6$. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

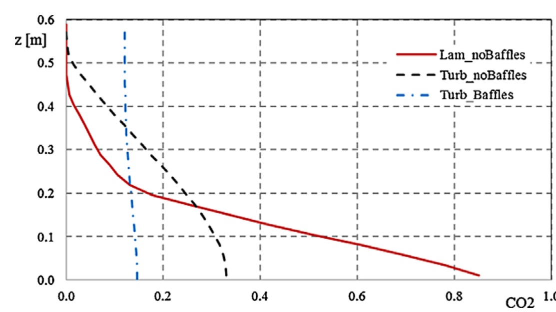


Figure 17 CO_2 concentration at the outlet section. Duct centerline corresponds to $z = 0$. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

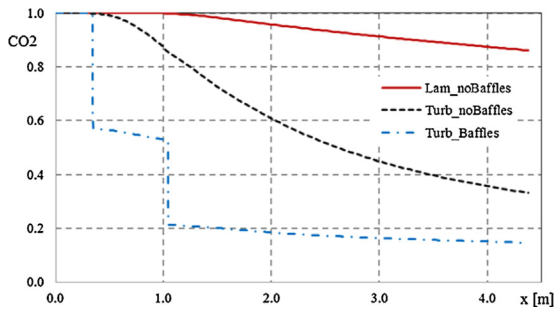


Figure 18 CO_2 concentration along the duct centerline. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

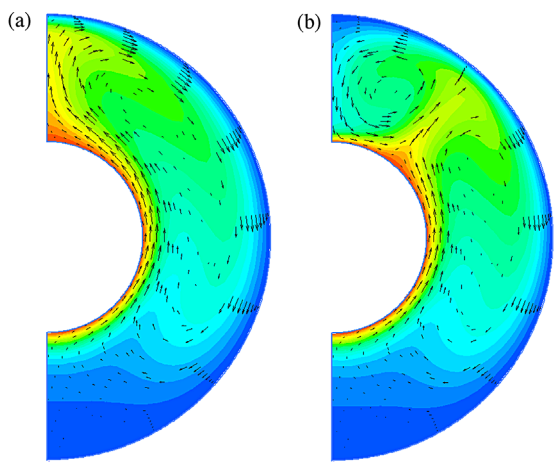


Figure 19 Temperature field for the natural convection inside a concentric annulus (the symmetric part is omitted). (a) Single vortex solution; (b) Two vortices solution. [Color figure can be viewed in the online issue, which is available at wileyonlinelibrary.com.]

Natural Convection Inside a Annular Cavity

This is a very interesting example of a bifurcation problem, which can be used to show to students how the same boundary conditions may lead to different solutions. Two concentric cylinders form an annular cavity filled with a fluid and subjected to a temperature difference. The problem is characterized by the Grashoff number, $Gr = 5\times10^4$, and by the Prandtl number, $Pr = 0.7$. Due to buoyancy effects, a natural convection laminar flow takes place, with ascending flow in the warmer inner cylinder and descending flow in the cooler outer cylinder, forming a symmetric pattern. Figure 19a illustrates this solution, with omission of the symmetric part. This corresponds to one of two possible solutions. The second solution is characterized by the presence of a secondary vortex, rotating in the opposite direction. The obtained solution (one- or two-vortices) depends on the choice of the numerical parameters for solving the problem, including sub-relaxation factors and solution initialization. This can be translated into the physical lab, where a perturbation on the flow may induce the second solution.

CONCLUSIONS

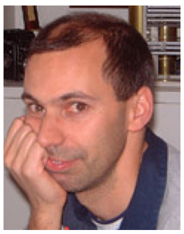
The CFD numerical tool EasyCFD_G was presented. This tool can be regarded as a very interesting numerical lab, where a wide range of physical situations involving fluid flow and heat transfer can be simulated. The easy and intuitive utilization were taken as leading criteria for the GUI design of this software, rendering it a useful tool for academic purposes. The software learning curve is fast and intuitive. Students can run a case, examine the solution, return to the problem definition to change some parameters or boundary conditions and check how the solution evolves. This provides the student a good way to understand the physics involved. A few example problems were presented in this work, along with comparisons with published solutions data for validation. Further to its educational value, the capability to simulate different types of physics, including turbulence, transient problems, mixing of fluids, and axisymmetric problems, makes EasyCFD_G a good alternative for analysis of applied cases were a 2D approach may be sufficient.

REFERENCES

[1] A. M. G. Lopes, A versatile software tool for the numerical simulation of fluid flow and heat transfer in simple geometries, Comput Appl Eng Educ 18 (2010), 14–27.
[2] B. E. Launder and D. B. Spalding, The numerical computation of turbulent flows, Comput Methods Appl Mech Eng 3 (1974), 269–289.
[3] B. E. Launder and D. B. Spalding, Mathematical models of turbulence. Academic Press, London and New York, ISBN 0-12-438050-6 1972.
[4] N. Djilali, I. Gartshore, and M. Salcudean, Calculation of convective heat transfer in recirculating turbulent flows using various near-wall turbulence models, Numerical Heat Transfer Part A 16 (1989), 189–212.
[5] F. R. Menter, Zonal two-equation k-w turbulence model for aerodynamic flows, AIAA Paper 1993-2906 (1993).
[6] S. V. Patankar, Numerical heat transfer and fluid flow. Hemisphere Publishing Corporation, Washington, D.C., ISBN 0-89116-522-3 1980.
[7] J. P. Van Doormaal and G. D. Raithby, Enhancements of the simple method for predicting incompressible fluid flows, Numerical Heat Transfer 7 (1984), 147–163.
[8] C. M. Rhie and W. L. Chow, Numerical study of the turbulent flow past an airfoil with trailing edge separation, AIAA J 21 (1983), 1525–1532.
[9] S. Majumdar, Role of underrelaxation in momentum interpolation for calculation of flow with nonstaggered grids, Numerical Heat Transfer 13 (1988), 125–132.
[10] W. Z. Shen, J. A. Michelsen, N. N. Sorensen , and J. N. Sorensen, An improved SIMPLEC method on collocated grids for steady and unsteady flow computations, Numerical Heat Transfer Part B 43 (2003), 221–239.
[11] H. K. Vertsteeg and W. Malalasekera, An Introduction to Computational Fluid Dynamics, Pearson Education Limited, 2007.
[12] L. W. Ehrlich, An ad hoc SOR method, J Comput Phys 44 (1981), 31–45.
[13] B. R. Hutchinson and G. D. Raithby, A multigrid method based on the additive correction strategy, Numerical Heat Transfer 9 (1986), 511–537.
[14] T. D. Blacker and M. B. Stephenson, Paving: A new approach to automated quadrilateral mesh generation, Int J Numerical Methods Fluids 32 (1991), 811–847.
[15] J. L. Steger and R. L. Sorenson, Automatic mesh-point clustering near a boundary in grid generation with elliptic partial differential equations, J Comput Phys 33 (1979), 405–410.

[16] P. D. Thomas and J. F. Middlecoff, Direct control of the grid point distribution in meshes generated by elliptic equations, AIAA J 18 (1979), 652–656.
[17] T. Nishimura and S. Matsune, Vortices and wall shear stresses in asymmetric and symmetric channels with sinusoidal wavy walls for pulsatile flow at low reynolds numbers, Int J Heat Fluid Flow 19 (1998), 583–593.
[18] J. Kim, P. Moin , and R. Mose, Turbulence statistics in fully developed channel flow at low reynolds number, J Fluid Mech 177 (1987), 133–166.
[19] I. H. Abbott and A. E. von Doenhoff, Theory of wing sections. Dover Publications, New York, 1959.
[20] C. L. Ladson, Effects of independent variation of mach and reynolds numbers on the low-speed aerodynamic characteristics of the NACA 0012 airfoil section, NASA TM 4074 (1988).

BIOGRAPHY



António Gameiro Lopes is Assistant Professor at the Department of Mechanical Engineering, Faculty of Sciences and Technology of the University of Coimbra, Portugal. Graduated in Mechanical Engineering in 1988; Diploma Course by the von Karman Institute for Fluid Dynamics, Belgium, in 1989; PhD by the University of Coimbra in 1994. Main teaching activities are in Fluid Dynamics, Computer Programming and Heat Transfer. Main research activities in Computational Fluid Dynamics and Heat Transfer, Aerodynamics and Forest Fires.