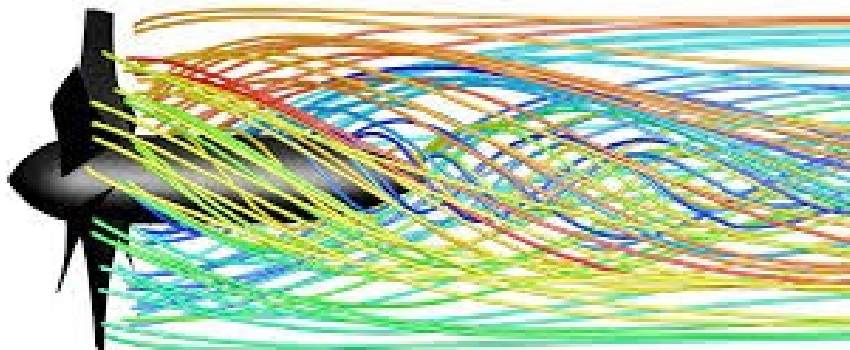


Application of flow visualization in numerical simulation



Students :

Théo DETIGE

Yiheng HAN

Hengshuo LI

Jincheng LI

Louise LEROUX

Éléonore SIMON

Project's coordinating teacher :

Mostafa Safdari SHADLOO

Date of report's submission : **15/06/2020**

Reference of the project : **STPI/P6/2020 – 028**

Title of the project : ***Application of flow visualization in numerical simulation***

Type of project : ***Simulation***

Goals of the project :

The main goal of this project is to discover and to get familiar with fluid flows as well as CFD software, which are often used by engineers in their professional lives. CFD stands for Computational Fluid Dynamics. It consists in solving numerically equations in the domain of fluid flow such as Bernoulli or Navier Stokes. More precisely, we chose Ansys Fluent to carry out our experiments. Another aspect of the project is the study of the different types of fluid flows and the equations which characterizes the latter. We could therefore establish a link between the theoretical part concerning the study of equations describing a fluid flow and the practical part, in which we could build a model of fluid flows.

Keywords of the project : ***Fluid flow, numerical simulation, CFD software.***

TABLE OF CONTENTS

1. Introduction.....	6
2. Description of our work.....	7
2.1. The steps followed.....	7
2.2. Specific project situation.....	7
2.3. Organization chart.....	8
3. Work accomplished and results.....	8
3.1. Software's choice.....	8
3.1.1. VisIt.....	8
3.1.2. Tecplot.....	9
3.1.3. Matlab.....	10
3.1.4. Paraview.....	10
3.1.5. Simscale.....	11
3.1.6. Ansys Fluent.....	11
3.1.7. Conclusion.....	12
3.2. Few information about the fluid flow and the CFD simulation.....	12
3.2.1. What is a fluid flow ?.....	12
3.2.2. Which type of resolutions ?.....	12
3.2.3. What is the CFD simulation ?.....	12
3.2.4. Which type of flow can be modeled ?.....	14
3.2.5. Different types of flow simulations.....	14
3.2.6. Different types of interface modeling.....	15
3.3. Modeling of an internal fluid inside an elbow.....	15
3.4. Modeling of an external compressible flow on an airfoil.....	17
3.5. Comparison between the two types of modeling and different post-processing.....	18
4. Conclusion and prospects.....	19
4.1. Conclusion for the results of our work.....	19
4.2. Prospects.....	19
4.3. Personal gain of this project.....	19

4.3.1. Théo.....	19
4.3.2. Yiheng.....	19
4.3.3. Hengshuo.....	19
4.3.4. Jincheng.....	20
4.3.5. Louise.....	20
4.3.6. Éléonore.....	20
5. Bibliography.....	21
6. Appendices.....	22
6.1. Airfoil's drag coefficient report.....	22
6.2. Airfoil's moment coefficient report.....	22
6.3. Airfoil's lift coefficient report.....	23
6.4. Airfoil's pressure coefficient plotted against the position in meters.....	23
6.5. Airfoil's X-Wall Shear Stress plotted against the position in meters.....	24

1. INTRODUCTION

In this research study, we have learned about the application of fluid flow visualization in numerical simulation with professor Mostafa Shadloo, as a group. There are six people in this group, we are working together under the guidance of the professor. Although there was a difficulty when we got to work with each other in distance, especially concerning the download of the software, we overcame it and got lots of achievements in the end.

The main purpose of this project is to get familiar with the domain of CFD (computational fluid dynamic) and initially grasp the using method of a CFD software. We can achieve this goal by following the steps below. It was very interesting and useful to discover such a theme since CFD software are frequently used by engineers in their jobs. We can split our approach of CFD software into two parts : a theoretical study and practical experiments.

For the theory part, we studied different software which concerns numerical analysis and data visualization and compare the differences, as well as, "Pros and cons". Indeed, we discovered a wide range of software, including *Tecplot*, *Paraview*, *Matlab*, *Simscale*, *VisIt* and *Ansys Fluent*. Through this first approach, we got familiar with the basic conception of CFD (Computational Fluid Dynamic), such as definition, purpose of use, analysis process...

Then, after we decided to focus on *Ansys Fluent*, we could deal with fluid flows a little bit more in depth. Indeed, we discovered the mathematical model for the flow modeling as well as the ways to treat different flows.

After a good theoretical approach, we could download the *Ansys* software on our computers in order to start the practical part of the project. We started by getting familiar to the interface of the software by trying to use it simply. Since the interface is quite user-friendly, we could quickly move on to the following part, which is to study an internal fluid flow by modeling an elbow in *Ansys Workbench*, exactly following the guidance of the professor. We carried on with the study of an external fluid flow by modeling an airfoil. We finished by executing the post processing for the running results in above situations.

In fact, we have already occupied some basic knowledge in flowing fluid in the first year. This is a particular chance for us to study it further with CFD software, it helps us understand deeply the mechanism of fluid flow by configuring an intuitive interface and several relevant parameter plots.

2. DESCRIPTION OF OUR WORK

2.1. The steps followed

In the beginning, our first task was to do some research on different CFD software; each member of the group had to find some information about a software, and if was possible, to do a simulation and then to do an oral presentation. The aim of this work was to find the appropriate software for the continuation of our project. Indeed, we chose the software *Ansys Fluent*.

As soon as we chose the software, we had to learn how to use it. The following task consisted in building a geometry in order to become familiar with the software. We had to set up a mixing elbow in the *Ansys Workbench* thanks to a tutorial given by our teacher and then, we had to study the fluid flow and the heat transfer within it. We were guided by the steps of the tutorial and we managed to create the elbow geometry and the corresponding computational mesh using the geometry and meshing tools within *Ansys Workbench*. Afterwards, we used *Ansys Fluent* to set up and solve the CFD problem and then visualized the results in *Ansys Fluent* and the CFD-Post Processing tools.

We also had to synthesize documents about fluid flows in order to learn more about CFD simulations and to compare it with a manual simulation.

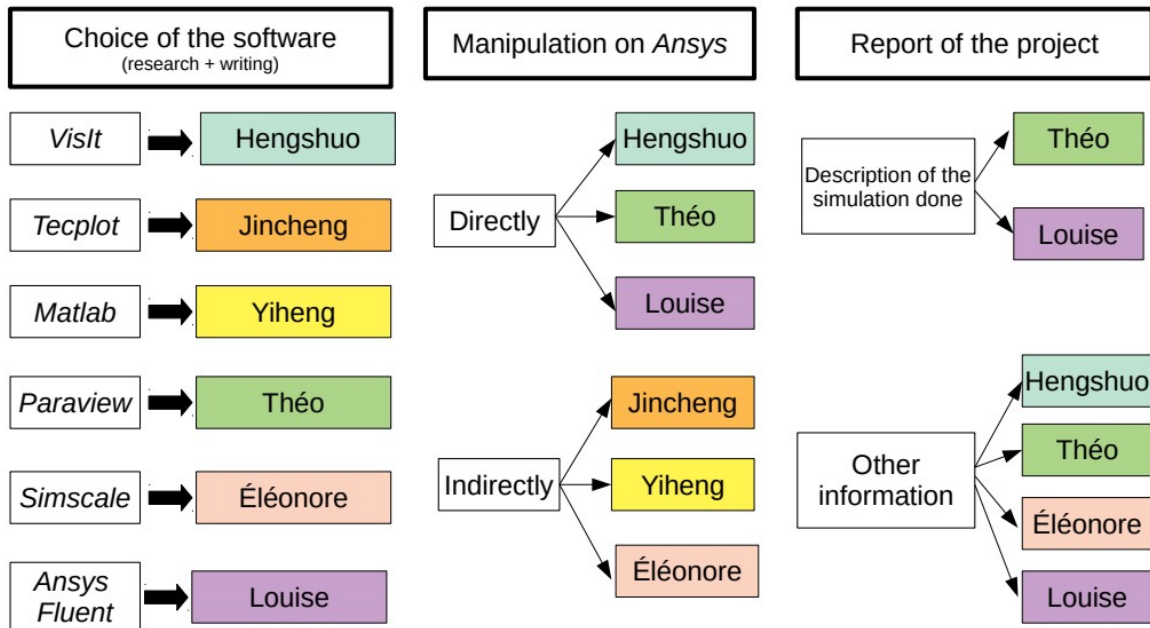
Finally, our last task resided in modeling an external compressible flow. The purpose of this tutorial was to compute the turbulent flow past a transonic airfoil at a nonzero angle of attack.

2.2. Specific project situation

Due to the lockdown, the project took place in another way, and we have encountered a few problems. Indeed, we didn't have the computer of the Insa to work on, so we had to download the *Ansys Fluent* software on our computer. Three members of the group succeeded in downloading the software so they could directly do the simulation, but the other members failed to install the software so they had to do the simulation indirectly.

In addition, we had frequent meetings on the platform Zoom, however it was sometimes hard for everyone to take part in the meeting since we were in different countries and some may have encountered connection problems. Despite all these hazards, we tried to carry out our project and to keep no one out.

2.3. Organization chart



3. WORK ACCOMPLISHED AND RESULTS

3.1. Software's choice

Each of us carried out a research on one particular software that we could use for our project. We found the advantages and disadvantages of each. We have obtained the following results.

3.1.1. *Visit*

As Open Source, *Visit* is an interactive and scalable tool for visualization, animation and analysis. By using *Visit* users can interactively visualize and also analyze data ranging from small scales ($<10^1$ core) to large ones ($>10^5$ core), so that there is numerous possible uses. Actually, it is able to handle large data set size in the terascale range, because *Visit* employs a distributed and parallel architecture so that the user can handle very large data sets interactively.

Visit is a very powerful analysis tool, since it has a rich feature set for tensor field, vector and scalar visualization. It provides support for derived fields which allow new fields to be calculated with the use of already-existing fields. On account of its good variety of computational meshes, this software can turn out to be essential for the user. Indeed, it supplies two- and three-dimensional point, curvilinear, rectilinear, and finally unstructured meshes.

Besides, *VisIt*'s graphical user interface is quite user-friendly, allowing both novice users to quickly get started visualizing their data, and power users access to advanced features. *VisIt* achieves extensibility through the use of dynamically loaded plugins. Every *VisIt*'s plots, operators, and database readers are implemented as plugins and are loaded at run-time from the plugin directory. New plugins can be added simply by installing them in this directory.

In my opinion, the disadvantage of *VisIt* is not so obvious, as a CFD software. I compared it with the *Ansys Fluent*, *VisIt* does not possess so many function options such as making a plot as *Fluent*, after all, it is free.

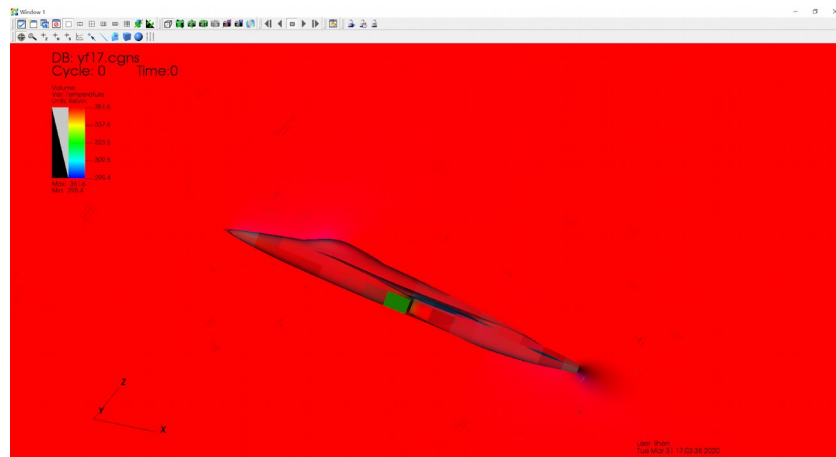


figure 1 : Airplane's simulation

3.1.2. Tecplot

Tecplot series software is a powerful data analysis and visualization software.

It provides a wealth of drawing formats, including x-y graphs, 2-D and 3-D surfaces in multiple formats drawing, and 3-D volume drawing formats. And the software is easy to learn and use, and the interface is friendly. And for *Fluent* software has a special data interface, you can directly read *.cas and *.dat files, you can also select the output surface and variables in *Fluent* software, and then directly output *tecplot* format documents.

Tecplot 360 is a numerical simulation and CFD visualization software that combines vital engineering drawings with advanced data visualization functions. It can quickly draw and generate animations based on data according to your ideas, analyze complex data, make various layout arrangements, and connect your results with professional images and animations. Of course *Tecplot 360* can also help you save time and effort in dealing with daily affairs.

Tecplot Focus is an engineering drawing software that allows users to draw all engineering data charts and test data charts as they like. With complete XY, 2D and 3D drawing functions, *Tecplot Focus* is designed for field data measurement, test data drawing, mathematical analysis, and overall project planning.

For engineers who want to understand the numerical simulation results of reservoir data faster, *Tecplot* is a professional visual analysis software that can help them improve efficiency. Users can choose to use line graphs, 2D cross-sectional views or 3D grid graphs for data analysis. It can be said that among similar post-processing software, *Tecplot RS* is the most easy-to-use software with the strongest visual output function. *Tecplot RS*

aggregates all your reservoir numerical simulation data and observation data together, allowing you to analyze and compare the data more quickly.

Such as other software, *Tecplot* has also some disadvantages, which are that it needs to be paid, and that we need a license to operate it too.

3.1.3. *Matlab*

Matlab is a combination of matrix and laboratory, which means matrix factory. It is a high-tech computing environment for scientific computing, visualization, and interactive programming released by mathworks. It will carry out numerical analysis, matrix calculation, scientific data visualization, modeling, simulation, and many other strong functions, which are embedded in an easy-to-use Windows environment. This software can be employed by scientific searchers or by design engineers. It must be effective to do numerical calculation in many other fields of science in order to provide a comprehensive solution, and to, at a great extent, get rid of the traditional non-interactive programming language (such as C and Fortran).

If we use *Matlab*, we have several advantages, which are quoted below. First, the programming is very efficient, since *Matlab* has the function of efficient numerical calculation and symbolic calculation, which can free users from complicated mathematical analysis. Moreover, the friendly user interface and the natural language close to mathematical expressions make it easy to learn and master. In addition, this software is fully functional, because it has the complete graphics processing function, which realizes the calculation result and the programming visualization. The application toolbox with rich functions (such as signal processing toolbox, communication toolbox etc.) provide the user with numerous convenient and practical processing tools.

Despite all of these advantages, *Matlab* also reveals disadvantages. Firstly the loop operation is inefficient, since all variables in *Matlab* are in vector forms, the process is quite complicated for a single element of a vector, or for a vector treated as a single loop variable. One solution is to exploit as many vector features as possible in the processing data. Secondly, there is a poor encapsulation : Although *Matlab* provides a certain amount of interactive interface production methods such as GUI, but as a real commodity, the final code will inevitably be ported to move 'low-level' languages, such as C or C++.

3.1.4. *Paraview*

Paraview is a multi-platform data analysis and visualization application. This software allows us to build quite easy visualizations in order to analyze various data. There are two options for the user to explore his data : either he can achieve it entirely in 3D or either he can choose to do it programmatically, with the use of *Paraview's* batch processing capabilities.

One of the major advantages of *Paraview* is that it was designed to analyze a huge and a large amount of dataets. In order to improve its efficiency, it can be run on supercomputers, which allows to analyze dataets with a petascale size. For more common and smaller data, it is also available on laptops. Although it might seem impressive for the first time, the *Paraview* interface is quite intuitive for the user. Moreover, the *Paraview* code base is very flexible. Indeed, every component of the code presents the opportunity to be reused in order to develop rapidly vertical applications. It is also possible for the user to create his own code from Python or C++.

The major disadvantage of such a software is that in 3D display, when a huge amount of data is treated, the screen display might be a little slow.

But despite this small drawback, this software has been distinguished a lot of times and has gained a solid reputation in international laboratories and universities.

3.1.5. *Simscale*

Simscale is a computer-aided engineering (CAE) software on which you can perform computer-aided design (CAD) models or computational fluid dynamics (CFD) models. The *Simscale* platform finds application in a wide range of industries, and is easy to use.

This software allows structural, fluid, particulate simulations, without the need of a powerful computer tool. Moreover, it is not necessary to download any software since you can directly simulate on the web site.

Simscale is not free but there is a version for students where you can simulate for free and grants you access to over 100,000 free engineering projects.

However, the current version of the *Simscale* platform only supports 3D workflows. Solving problems of lower dimension is in the internal testing phase and will be available soon.

Here are some screenshots of a simulation I made on *Simscale*, representing a pipe junction flow :

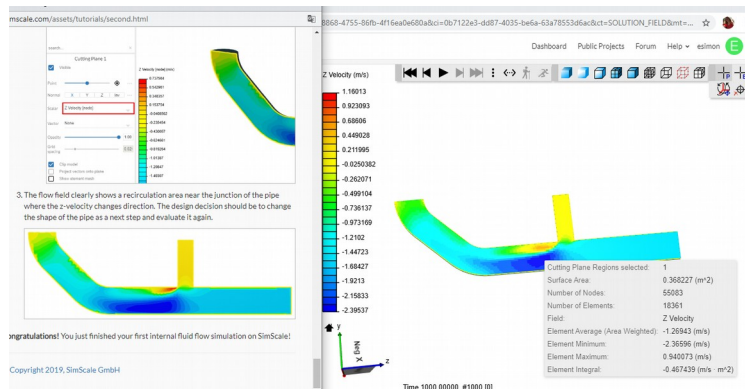


figure 2 : Pipe junction's simulation

3.1.6. *Ansys Fluent*

Ansys Fluent makes Computational Fluid Dynamics (CFD) simulation, that is to say that it provides fast results for modeling fluid flows, turbulence, heat transfer, chemical reactions, etc... It is used in a wide range of industrial applications, and in many companies.

Moreover, this software is quite user-friendly, since it enables the user to work directly with User Defined Functions (UDF), that he doesn't have to program before. If the user wants, he can also program the whole steps of the simulation, including the geometry, the mesh, the setup and the solution. The language employed by *Ansys* is the C programming language. Since the user has the choice to directly use UDF or to program the whole, *Ansys* is designed for both beginners and experts.

Besides, it has a clear graphical interface with a single window workflow and with task-based customization, which simplifies the user's choice, because he can directly enter expressions. The simplicity of this software does not mean that it is not an efficient software. Indeed, the results are very accurate and the calculation is speedier than in other software.

Concerning the cost of *Ansys Fluent*, there is a free version for students, but otherwise it is a costly software. Furthermore, it is installed on a few INSA's computers.

The only disadvantage of *Ansys Fluent* is that there are sometimes problems with tolerant values used to terminate iterative processes (such as for the convergence for example).

3.1.7. Conclusion

Thanks to our research we discovered what type of simulation software it exists, how they work, and which advantages and disadvantages they present. To conclude, we decided to choose the last software, i.e *Ansys Fluent*, to carry out our project, since it is a free software, it is available on many INSA's computers and it is moreover user-friendly.

3.2. Few information about the fluid flow and the CFD simulation

3.2.1. What is a fluid flow ?

It involves the motion of a fluid, when the latter is subjected to unstable forces. We can see fluid flows in everyday life such as in meteorological phenomena or with objects in interaction with water or air. Indeed, during an interaction of any object with water (such as a boat, an animal...) or air (wind turbines, airplane, birds...). The fluid flows are studied in many domains including environmental hazards, heating, ventilation and air conditioning, combustion in propulsion systems or processes in the human body such as blood or breathing.

3.2.2. Which type of resolutions ?

To understand how the fluid reacts and where it comes from, we can find analytical solutions thanks to mathematical equations (among other the Navier-Stokes equations), as well as experimental and numerical solutions. Then, it discretizes these equations into an algebra problem by the use of Taylor series. After, the problem is solved and validated, which allows us to achieve the simulation.

3.2.3. What is the CFD simulation ?

CFD means Computational Fluid Dynamics, it gathers computer simulation for prediction of fluid-flow phenomena with the need of some algebraic equations (Euler, Stokes...)

It is a powerful tool since it enables scientists and engineers to perform numerical experiments with computer simulations, which can be qualified as "virtual laboratory". Thanks to CFD, a real experiment can be more accurate and so can be analyzed easier. Because of this, CFD enables searchers and engineers to simulate experiments virtually in a lot of fields, like chemistry, meteorology, military or architecture.

CFD analyzes a fluid-flow phenomenon in several steps : first of all, it realizes a model of the continuous fluids using Partial Differential Equations. Then, it discretizes these equations into an algebra problem by the use of Taylor series. After, the problem is solved and validated, which allows us to achieve the simulation.

CFD involves many disciplines in the research area, such as physics, applied mathematics and computer science. However, a typical CFD cycle is quite simple to resume. At the beginning, the analyst states the problem. Then scientific knowledge and computer code allows the computer to realize complex calculations, namely the discretization process, and thanks to an iterative solver they provide a solution. The latter is then interpreted by a specialist.

The essential part for the computer operation is to check the residuals, relative solution changes and other indicators to make sure that the iterations converge. Solution is converges, residuals do not change after more iterations.

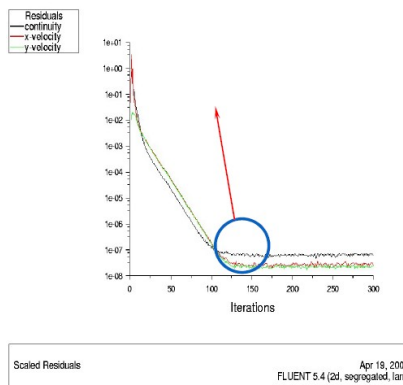


figure 3 : Residuals' graph (plotted against iterations)

After getting the solution, we can plot derived quantities (stream function, vorticity), as well as integral parameters (total mass, lift, drag), in 1D data (function values connected by straight lines), in 2D data (streamlines, contour levels, color diagrams), or in 3D data (cut lines, cut planes, isosurfaces, isovolumes). We can also plot arrow, particle tracing,s, animations, etc...

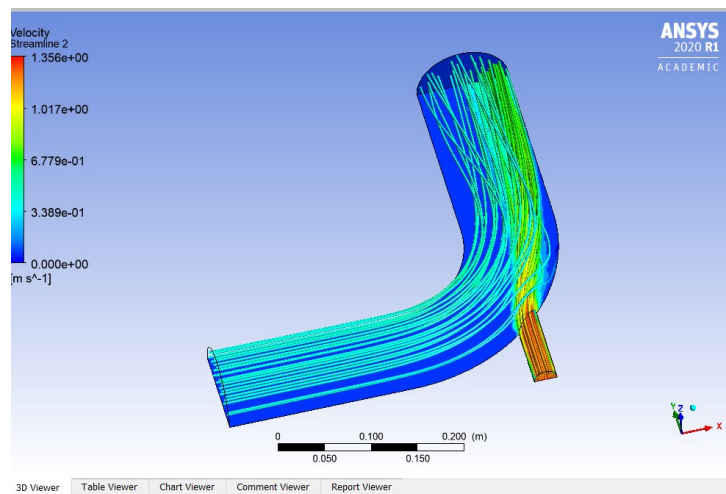


figure 4 : Velocity streamlines inside the elbow

Before CFD appeared, there were two ways to analyze the fluid flow : the Analytical Fluid Dynamics (AFD) and the Experimental Fluid Dynamics (EFD). Unlike traditional experiments that give a description, CFD simulation gives a prediction of flow phenomena for all desired quantities. It is also cheaper and faster compared to a traditional experiment since you don't have to do some measurements, you just have to simulate on the CFD software. In addition, some fluid phenomena are sometimes difficult to be measured by experiments and easier to simulate by CFD. Otherwise, CFD is more precise for laminar, slow and single-phase flow.

However, the results of a CFD simulation are never totally reliable due to the limited accuracy of the result or the imprecision of the input data. Although CFD is apparently a very efficient way to analyze fluid-flow phenomenon, it also may be the cause of some errors or uncertainty. The errors might appear during phases of modeling, discretization, iteration or implementation. For instance, the physical model might be false due to uncertainty and simplifications realized by the analyst. The approximation of Partial Differential Equations by algebraic equations might cause a discretization error. During the iteration, the stopping criteria might be at the origin of an approximation causing an error in the final solution. All those errors are made on purpose, and have to be taken in consideration in the final analysis of the result. That's why the results of CFD simulation are never 100% reliable because of rounded or imprecise data. But there are also some errors that can't be detected, such as errors in the computer programming or wrong parameters values, models or boundary conditions used by the analyst.

3.2.4. Which type of flow can be modeled ?

There are two main types of flow, that is to say the turbulence and the laminar flows. Both can be solved with whether the Eulerian or the Lagrangian or the mixed Eulerian-Lagrangian coordinate system. Never mind what type of coordinate system we choose, because they provide the same equations. Laminar flows are defined by parallel and regular layers. On the contrary, the fluid motion of turbulence flows are illustrated by disorganized changes in pressure and velocity. That's why it gives prominence to the irregularity, the diffusivity, the rotationally, and the dissipation of the fluid. To simulate these turbulent flows, we can use either a Direct Numerical Simulation (DNS) or a Large Eddy Simulation (LES). The first one needs to settle the solutions of the Navier-Stokes equations for the whole spatial and temporal scale, that is to say that it doesn't need any turbulence model. The second one is a mathematical model for turbulence, thus being less costly in calculations because it ignores the smallest length scales.

3.2.5. Different types of flow simulations

There are three main types of turbulent flows simulations. First, the Direct Numerical Simulations (DNS) allows to solve numerically Navier-Stokes equations without the use of any turbulence model, that is to say that the full range of spatial and temporal scales of the turbulence have to be solved. The second type of flow simulation is the Large Eddy Simulation (LES). It is a mathematical model for turbulence and permits to reduce the computational cost. Indeed, it ignores the smallest length scales using low-pass filtering of NSE. The last type is the Reynolds-averaged Navier-Stokes equations (RANS). The idea behind the equations is Reynolds decomposition, whereby an instantaneous quantity is decomposed into its time-averaged and fluctuating quantities, an idea first proposed by Osborne Reynolds. These equations can be used with approximations based on knowledge of the properties of flow turbulence to give approximate time-averaged solutions to the Navier-Stokes equations.

3.2.6. *Different types of interface modeling*

When we consider an Eulerian description of the fluid-flow, there are different types of graphs are used to represent the results of the simulations. The interface-capturing is a simple discontinuity line of a characteristic function. The surface-tracking represents the interface by various series of interpolated curves through a discrete set of points. Finally, the volume-tracking reconstructs the interface when necessary and doesn't store its representation. Here, the important point is the marker quantity within the cell.

When we focus on a Lagrangian description, two methods exists for the interface modeling of the fluid flow. First, the Free Lagrangian Methods (FEM) allows the originally represented mesh to move and deform. From this point, a new mesh is built in conjugate with the scrambled mesh. The other method is the Meshless particle methods (SPH). Here, the grid is entirely abandoned. That's why the flow is represented by replacing the original mesh with a finite number of particles. These particles carry many of the fluid characteristic properties, including its velocity, position, or mass. Therefore, the fluid system evolution is governed by interactions between these particles.

Finally, it is possible to mix both Eulerian and Lagrangian methods. This leads to two different treatments. The first one is the Segregated flow-interface treatment (ALE). It first consists of a Lagrangian calculation which doesn't move the mesh vertices. Then, an implicit adaptation of iterative velocity and pressure field is followed by the mesh vertices movement towards their new position. The treatment ends with a mesh rezoning for a new configuration. The second method which mixes Eulerian and Lagrangian descriptions is the Integrated flow-interface treatment. Here, the flow equations are discretized with their corresponding interfacial boundary conditions. The nonlinear algebraic equations are therefore solved by any iterative procedure such as Newton or quasi-Newton ones.

3.3. **Modeling of an internal fluid inside an elbow**

We discovered the *Ansys Fluent* fluid flow systems and *Ansys Workbench* by setting up and solving a three-dimensional turbulent fluid-flow and heat-transfer problem in a mixing elbow. It is a quite simple geometry, allowing us to familiarize with the *Ansys Workbench* tool set. In this part, we created an elbow geometry and its corresponding computational mesh with *Ansys Workbench*.

Then, we started using *Ansys Fluent* in order to set up and solve the CFD problem. Finally, we visualized the results in *Ansys Fluent* and the CFD-Post post processing tool. We study a cold fluid at 293.15 K flowing into a pipe through a large inlet. This fluid comes to mix with a warmer fluid that has a temperature of 313.15 K and which is coming in through a smaller inlet positioned on the elbow.

We often encounter this type of configuration in piping systems in power plants but also in process industries. For a good design of the junction, it is necessary to predict the flow field and temperature field in the area of the mixing region. We began by creating a *Fluent* fluid flow analysis system. After naming the analysis and saving the project, a list of files is generated by *Ansys Workbench*, displaying the name and type of file, the ID of the cell associated with the file and its location.

We can now start to create the elbow geometry. In order to do so, we have several options : either we can use *Ansys DesignModeler* or *Ansys SpaceClaim Direct Modeler*, or even directly import the adapted geometry file. The geometry that we intend to create here is a large curved pipe accompanied by a smaller side pipe. We begin with the creation of the main pipe by defining a segment of a torus in base Y direction vector. Then, we extrude the

faces of the torus segment in order to create straight inlet and outlet lengths. After this step, we design the side pipe by adding a cylinder primitive. Finally, we reduce the model we just created to half of the pipe assembly by using the symmetry tool, in order to reduce computational cost.

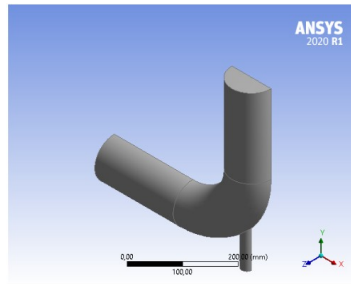


figure 5 : Elbow's geometry

Once the mixing elbow geometry is created, we then have to generate a computational mesh throughout the flow volume. During this step, we will use the *Ansys* Meshing application.

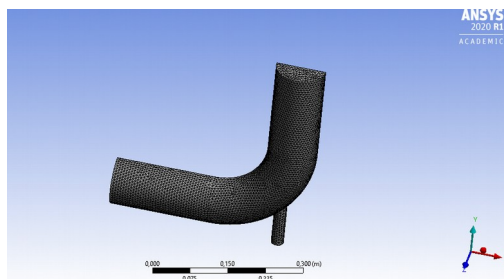


figure 6 : Elbow's mesh

When the creation of the computational mesh is completed, we set up a CFD analysis with *Ansys Fluent*. This step consists in setting the physics parameters before solving the problem by setting up and running the calculation. Finally, we can display the results of the simulation in *Ansys Fluent* and *CFD-Post*. We displayed the velocity contour in our two elbows.

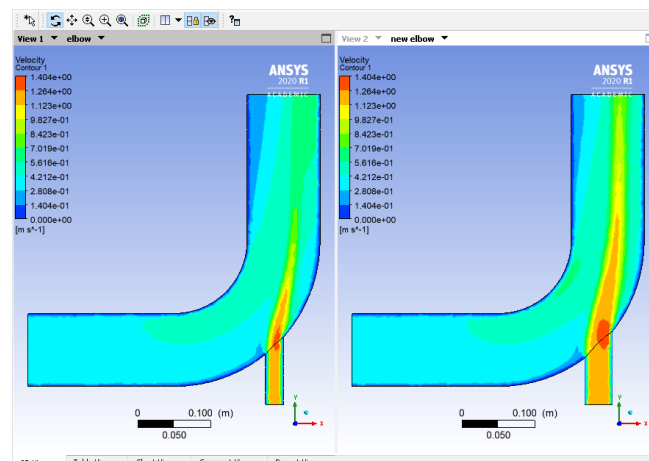


figure 7 : Velocity contours in the 2 elbows

3.4. Modeling of an external compressible flow on an airfoil

After the modeling of an internal fluid flow inside an elbow, we now try to compute the external compressible turbulent flow past a transonic airfoil at a nonzero angle of attack using the Spalart-Allmaras turbulence model. So we will study the flow around an airfoil at a 4°-angle of attack and a free stream Mach number of 0.8. Since it is a transonic flow, its speed is close to that of sound and the flow has a strong shock near the mid-chord on the upper side. Below is a representation of the problem description.



figure 8 : Problem statement of the airfoil

The airfoil's mesh is given by our teacher, so that we can instantly set it up. In the *Ansys Workbench*, we first open the "Setup" menu, and import the mesh there. Afterwards, it is recommended to check once the mesh, to control if the reported minimum volume is a positive number. If it wasn't the case, that would mean that the mesh is incorrect, and then it is recommended to change the mesh' parameters or to modify the geometry. Here is the zoomed obtained mesh.

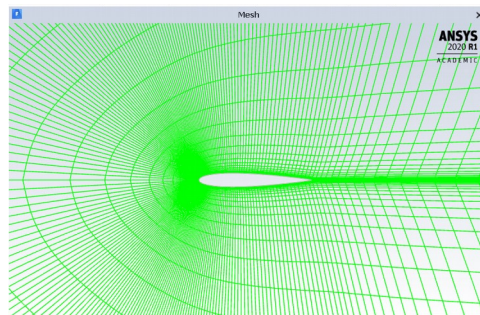


figure 9 : Airfoil's mesh

Once the mesh is approved, we can solve it. First we have to set up some physic values with the Spalart-Allmaras model, the fluid material (air), and to decide on the boundary conditions for external aerodynamics, which also includes the operating conditions : the gauge pressure amounts to 0 Pa, the Mach number to 0.8, the temperature to 300K, the operating pressure to 101325 Pa and, since we want a 4°-angle of attack, the X-component direction and the Y-component direction are 0.997564 and 0.069756 respectively. Afterwards we can get the solution by setting first the solution parameters (methods), and then the controls (density of 0.5 and modified turbulent viscosity of 0.9). To plot the final calculation, we just need to enable some residual plotting in the residual monitors, and subsequently to initialize the solution. In this modeling we choose to save the case and data during several stages of the setup before calculation. In order that the computer can have some reference values to non-dimensionalize the forces and the moments acting on the airfoil, the user puts them on the "reference values" menu. Therefore the user creates a force report definition to plot and write the coefficient simultaneously for the walls of the airfoil, for the lift coefficient, and also for the moment coefficient.

We now come to the post-processing by editing some contours that we want to visualize, that is to say the pressure.

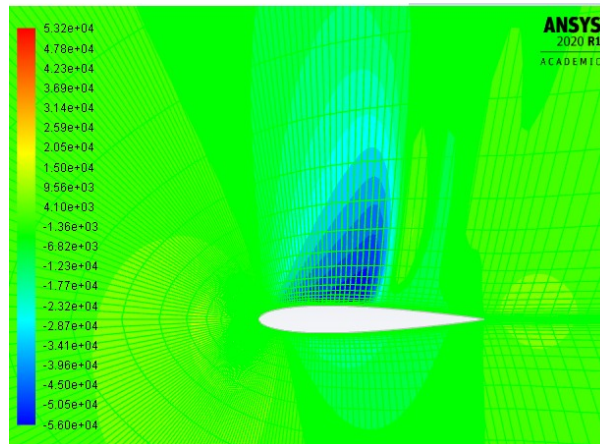


figure 10 : Contours of static pressure on the airfoil

The shock on the airfoil is clearly noticeable on the wall-top, where the pressure raises to a higher value than in another area. To analyze more precisely what happened in this area, we create a point just downstream of the shock wave to define a new surface report (vertex average report type). Once we do that, we can run again the calculation to draw the graphs of the vertex average of velocity magnitude, the drag, the lift, and the moment plotted against the number of iterations. The graphs of each of them are attached in the appendices.

In the post-processing, we can also plot the y^+ distribution on the airfoil by editing the “XY plot” menu. We choose the Y Axis function and the X Axis function : here it is first the pressure coefficient and then the X-Wall Shear Stress, which are plotted against the position in meters (refer to the appendices).

Finally, some vectors can be drawn thanks to the post-processing, and we did it for the velocity vectors downstream of the shock.

In conclusion, thanks to this tutorial we have learned to solve an external aerodynamics problem using the pressure-based coupled solver with some turbulent models, to observe the monitor convergence using force and surface report definition and to discover the different uses of the post-processing.

3.5. Comparison between the two types of modeling and different post-processing

Once we have set up the both modeling, we can plot different things that we want to examine. We have first tried to adopt the post-processing of either cases for the other one. Then we plotted in each case some other things that we haven’t plotted yet. We have discovered the Q-criterion on an isosurface, which was explained after by our teacher. Moreover he showed us how to plot a streamline, so that we can good visualize the compartment of the flow when there is a mixing, i.e when there are 2 or more inlets inside a system (it was here the example of the elbow). The mixing depends indeed on the diameter of the inlets.

4. CONCLUSION AND PROSPECTS

4.1. Conclusion for the results of our work

Most of the purposes of this project are achieved, except the fact that the theory part is hard to understand for us now. What we just understand are some PDEs (Partial Derivative Equations) and initially grasp some of the models chosen for the turbulent flow. In the practical part, we strictly followed the tutorial which the professor sent us and made it step by step. It is not quite hard but interesting, fruitful in the process.

4.2. Prospects

The project opens the gate of numerical simulation domain for us, paving the way for our future study. Most of us choose Meca or EP as our major, the CFD will be an indispensable part for the future, and naturally, the earlier you learn, the better it is.

4.3. Personal gain of this project

4.3.1. *Théo*

Discovering such a software has been very interesting since I had no experience in this type of software. Moreover, since I consider seriously to pursue my engineering studies in the Energetic and Propulsion department, I think that being initiated to this type of software will be very useful for me next year. Moreover, it is interesting to discover a concrete tool that is really used by engineers in their work. Even though I don't really feel like spending too much time on a computer, I know that it is a very important tool for an engineer and to my mind, it is useful to spend time to familiarize with numerical applications that are used very commonly in the field of engineering. Although I already had to write some reports in my studies, I always consider it as a difficult exercise because it's complicated for me to summarize my work. That's why the quite short length we had to produce to present our work helped me to work on this point. Last but not least, working in a group is always interesting and useful for our future jobs. This situation was even more interesting with the working conditions we had to deal with and helped me familiarize with virtual meetings and working from home.

4.3.2. *Yiheng*

In this project, first of all, I learned about the use methods and functions of these several software, some of which I had never heard of before. In addition, when using CFD to visualize the fluid, I learned to use the previous knowledge reasonably in the data simulation. Finally, I felt a sense of achievement when I witnessed the results with my group. I believe that if I can be engaged in related industries in the future, the experience of this project is very important.

4.3.3. *Hengshuo*

I have learned how to use a CFD with certain process, model choose for different situation (initially), the mathematical knowledge behind CFD function and also use

DesignModeler to create a schema of an object. In fact, it gives me the curiosity in this domain, I'm really interested in seeing the various parameter visualizing in the interface. It is very intuitive and sometimes beautiful.

4.3.4. Jincheng

This project enables me to get to know these software which may have the great chance to be used in my future years of study and also of work. After careful selection and discussion, we finally chose to use *Ansys Fluent*. In the process of using the CFD function to visualize the fluid, I have learned how to apply what I have learned in practice. Also it helps me to understand the behaviour of the fluid more easily. What's more this project gives me a great opportunity to cooperate with each other. To my future life, this project is rather important and helpful.

4.3.5. Louise

This project has enabled me to discover a new type of software, which is the simulation one. At the beginning of the project, when each of us researched on one particular software, I heard new software names that we will certainly use later. By the choice of *Ansys Fluent*, I could really deepen my view on CFD simulation, and succeed in using it. Besides, I have learned a lot of things concerning the behavior of a flow inside and outside a system, and how much it is complicated to solve it without the iterative solver of a computer. This project sounded undoubtedly important to me, because at first sight the invisible fluid flow is everywhere present in our lives, and secondly, energetically spoken, it can be one of the first causes of wasted energy, i.e of pollution. Since I would like to work in this domain, this project has been truly interesting. Finally due to the actual exceptional situation, I have also learned how to manage to work together virtually, and this can be a great experience for the future.

4.3.6. Éléonore

First of all, this project allowed me to discover the field of numerical simulation and brought me more knowledge about fluid flows. Indeed, the first research on CFD software introduced me to these software that I did not know or whose name I had only heard, this knowledge could help me in my future life since these software are mainly used in various job fields. Moreover, using *Ansys Fluent* abetted me getting familiar with numerical simulation and exploiting results in a different platform, and on top of that it also allowed me to manipulate a second modeling software since I also use *SolidWorks* for another project. Finally, working in a group with foreign students encouraged me to improve my English and made me project into a group of engineers, since videoconferencing is a common way of working.

5. BIBLIOGRAPHY

[1] Mostafa Shadari Shadloo, "Lecture 1", "Lecture 2", "Lecture 3", April 2020.

[2] Internet links :

-<https://www.ansys.com/> (valid on the date of 12/06/2020).

-<https://www.afs.enea.it/project/neptunius/docs/fluent/html/udf/node252.htm> (valid on the date of 10/02/2020).

-https://www.simscale.com/?fbclid=IwAR3_y3bVxarLtT_tXfwOmWj3BOxkZ1CY3qtdArEuqR66jmBTGW77P9icrx8 (valid on the date of 12/06/2020)

-<https://www.g2.com/products/simscale/reviews?fbclid=IwAR25q448BRhLjXQ1bQRP-xDyhRqUkyldJAiBf00-goqVH19Z9q2FUJa4Wt8> (valid on the date of 12/06/2020)

-<https://fr.mathworks.com/products/matlab.html> (valid on the date of 12/06/2020)

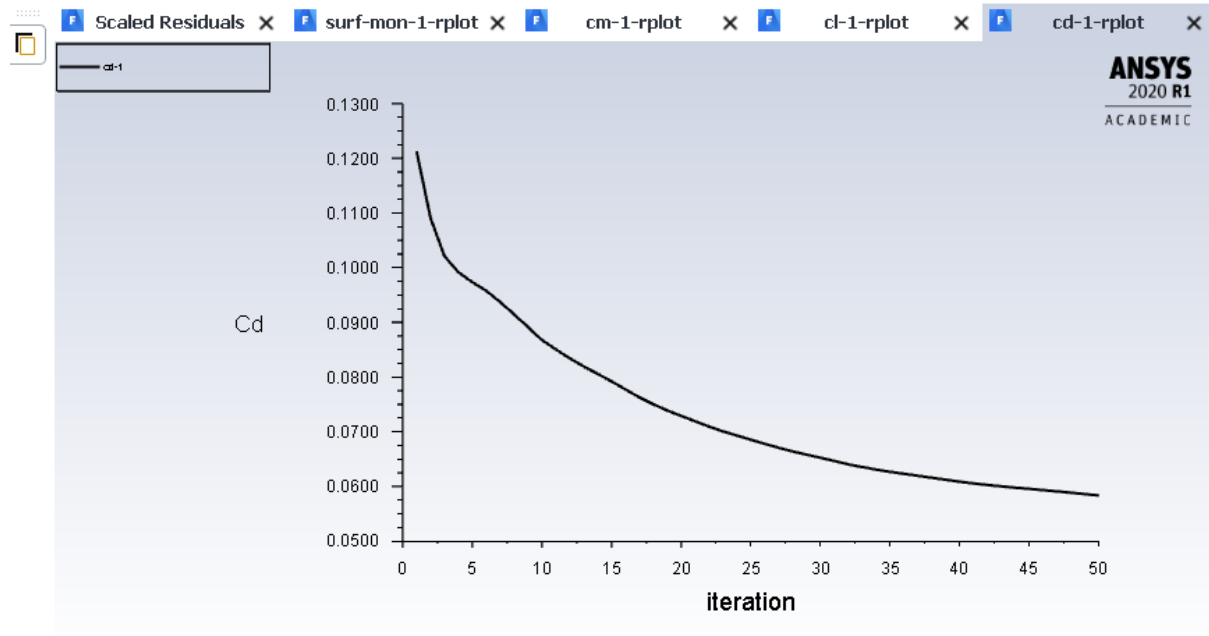
-<https://www.tecplot.com/> (valid on the date of 12/06/2020)

-<https://www.paraview.org/> (valid on the date of 12/06/2020)

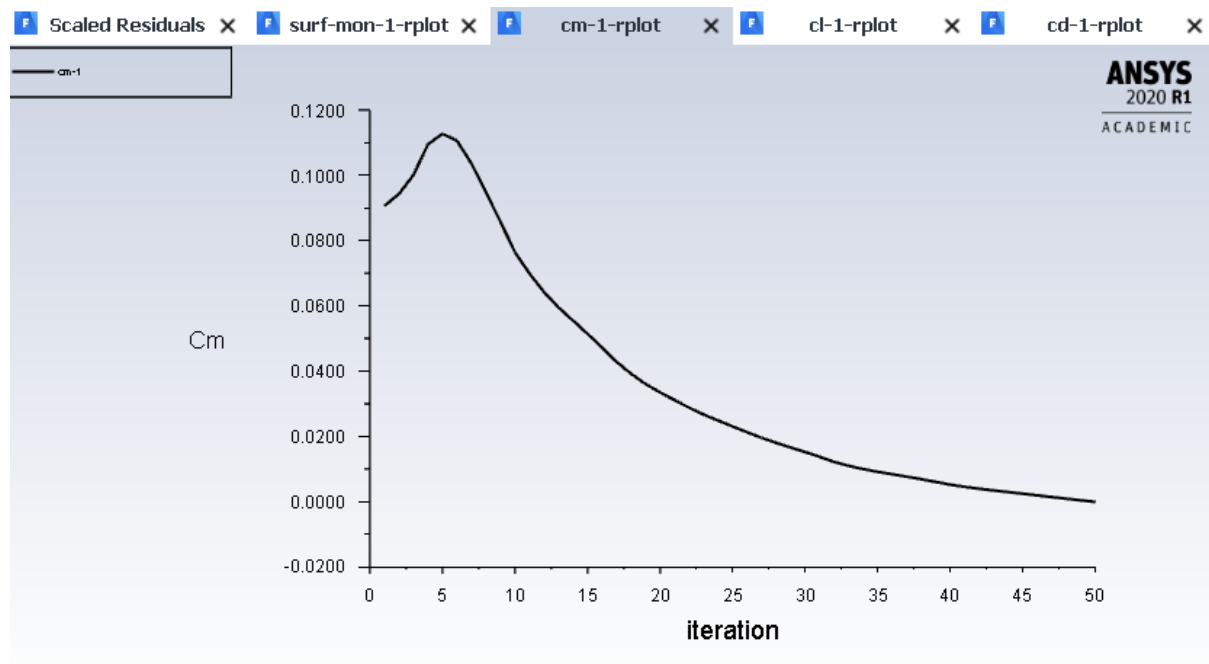
- <https://wci.llnl.gov/simulation/computer-codes/visit/> (valid on the date of 12/06/2020)

6. APPENDICES

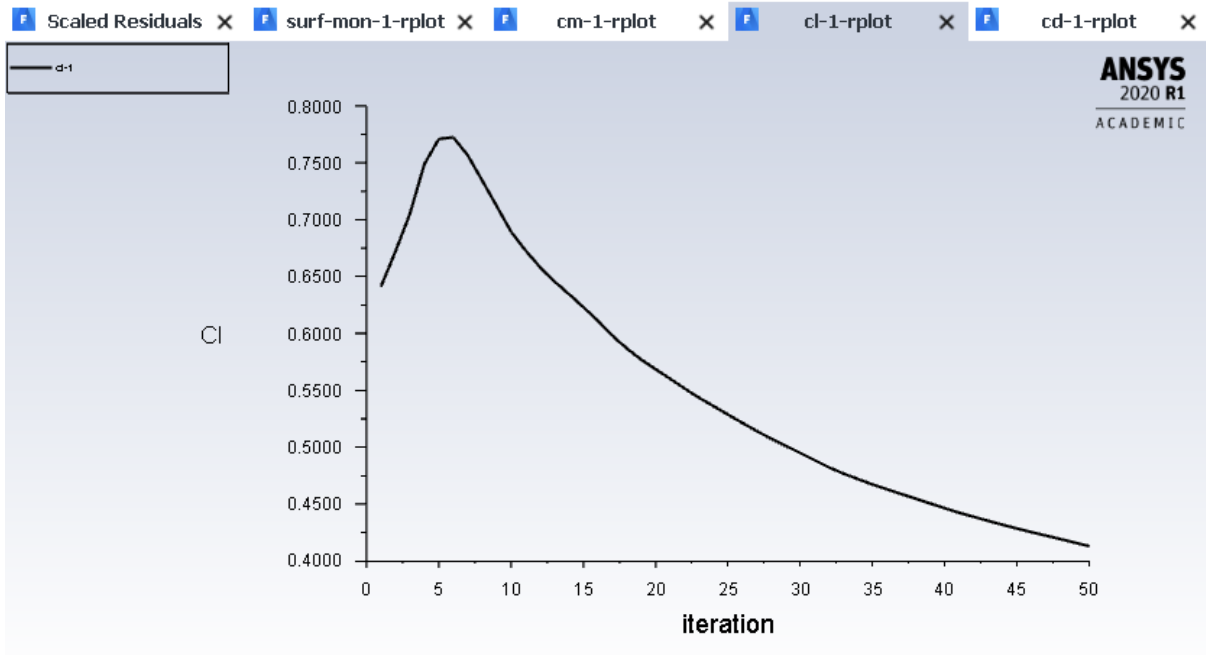
6.1. Airfoil's drag coefficient report



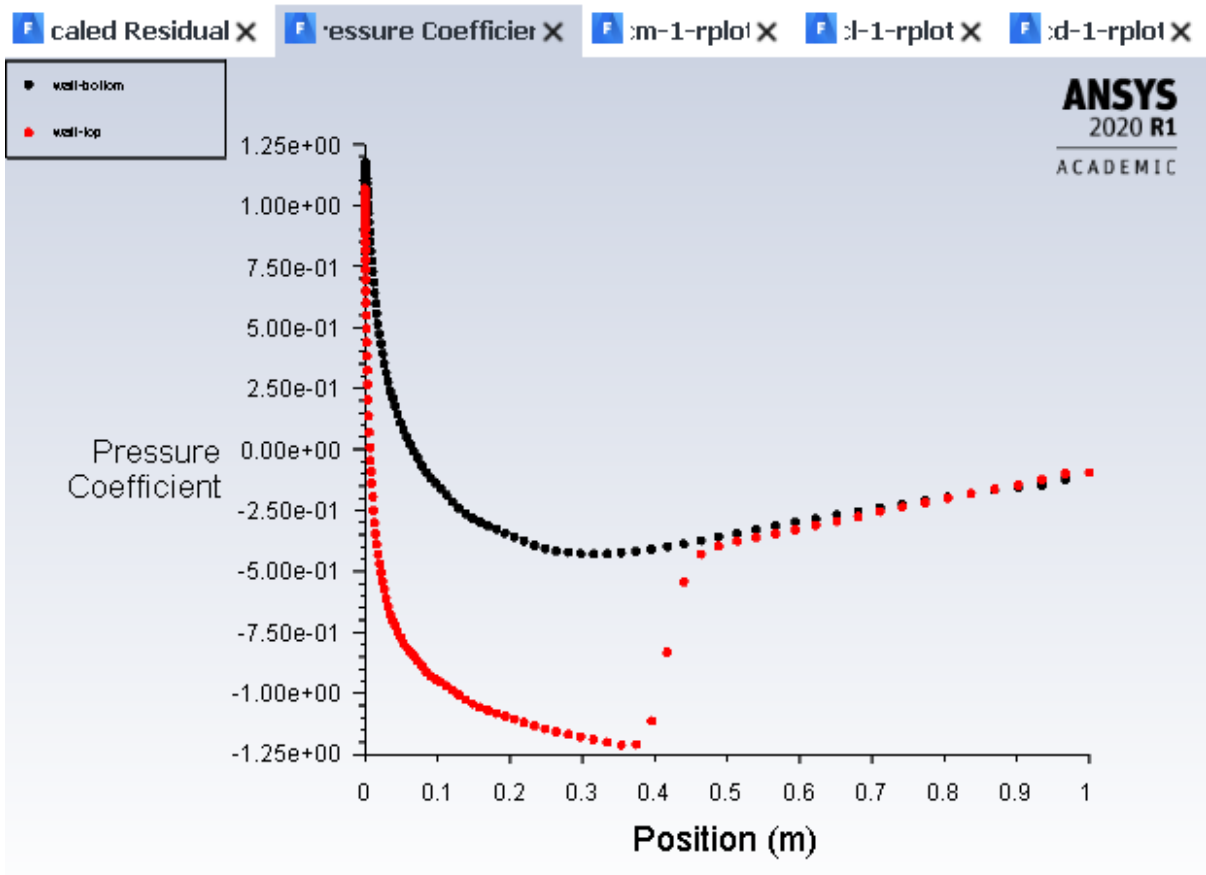
6.2. Airfoil's moment coefficient report



6.3. Airfoil's lift coefficient report



6.4. Airfoil's pressure coefficient plotted against the position in meters



6.5. Airfoil's X-Wall Shear Stress plotted against the position in meters

