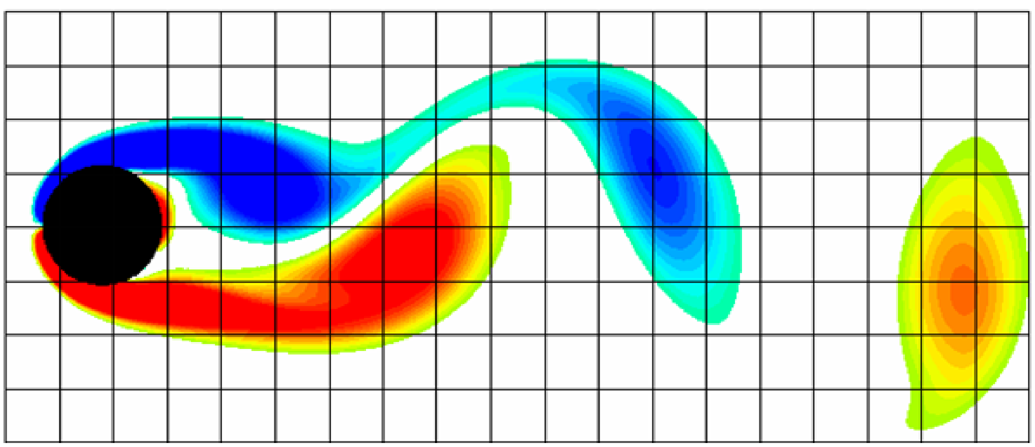
COMPUTATIONAL FLUID DYNAMICS

Exercise book - Master ENERGY



Olivier BOTELLA

LEMTA – Université de Lorraine - CNRS
2 avenue de la Forêt de Haye, TSA 60604
54518 Vandoeuvre (France)

Email : Olivier.Botella@univ-lorraine.fr

Lectures web page : http://arche.univ-lorraine.fr/course/view.php?id=9396

Lectures on Computational Fluid Dynamics (CFD)

Instructor : Olivier Botella

Master SRE, KazNU, March 2016

I. Basics of CFD and the finite-difference method (FDM) (4 days -16 h)

1) Day 1 : Introduction to CFD analysis - Partial Differential Equations (PDEs) for fluid flow problems - Finite Difference Method (FDM) (4h lecture)
2) Day 2: FDM for unsteady PDEs I: the heat equation. (2h lecture – 2h exercises)
3) Day 3: FDM for steady PDES: the convection-diffusion equation (1h lecture- 3h exercises)
4) Day 4: FDM for unsteady PDEs II: the transport equation. (2h lecture – 2h exercises)

II. Finite-volume method (FVM) for CFD (3 days- 12h)

1) Day 5: FVM for unsteady convection-diffusion equation. (2h lecture – 2h exercises)
2) Day 6: Meshes for CFD. FVM on unstructured meshes. (2h lecture – 2h exercises)
3) Day7: Solution of the Navier-Stokes equations. Introduction CFD analysis with ANSYS FLUENT software. (4h Lecture)

III. CFD lab exercises with ANSYS FLUENT. (3 days – 12h)

1) Day 8: Expansion flow in a pipe I: geometry & grid generation. (4h on PC)
2) Day 9: Expansion flow in a pipe II: flow computation in laminar and turbulent regimes (4h on PC).
3) Day 10: Flow pas circular cylinder (4h on PC).

IV. Day 11 : Written examination (2h).

A I. Basics of CFD and the finite-difference method (FDM)

A.1 Finite Difference (FD) Method I : The Basics

Exercice A.1 (FD formulas on uniform mesh).

We consider the uniform mesh $\{x_i = i\Delta x\}$ where Δx is the grid size.

1) Use centered FD formulas to approximate $u'(x_i)$ and $u''(x_i)$ for the following functions : $u(x) = 1$, x and x^2. Compare with analytical results. What would happen with polynomials of higher degree ?
2) By using Taylor expansion, show that the 3-point centered FD formula :

$$\delta^0_{xx} u_i = \frac{u_{i+1} - 2\,u_i + u_{i-1}}{\Delta x^2}, \tag{A.1}$$

is a second order approximation of $\left(\frac{\partial^2 u}{\partial x^2}\right)_i$.

3) Calculate the FD formulas $\delta^0_x \delta^0_x u_i$ and $\delta^-_x \delta^+_x u_i$. In what case the result is identical to the usual FD formula $\delta^0_{xx} u_i$ for the second derivative ? Calculate the second derivative of the 2 discrete functions $u_i = 1$ and $u_i = (-1)^i$ with these 2 formulas and comment the results.
4) Use the general technique to build the second-order forward scheme which computes $(\partial u/\partial x)_i$ with the values u_i, u_{i+1}, and u_{i+2}. Show that the truncation error reads :

$$\tau_i = -\frac{\Delta x^2}{3}\left(\frac{\partial^3 u}{\partial x^3}\right)_i + \mathcal{O}(\Delta x^3).$$

5) We are often faced with the problem of determining the truncation order p of a given FD scheme, which can also be useful to check for implementation issues when p is known. The standard way to obtain the actual order p of the scheme is to halve the parameter Δx and to calculate the reduction factor R, which is defined as the ratio between the truncation errors $\tau_i^{\Delta x}$ and $\tau_i^{\Delta x/2}$. Show that :

$$R = \frac{\tau_i^h}{\tau_i^{h/2}} \simeq 2^p. \tag{A.2}$$

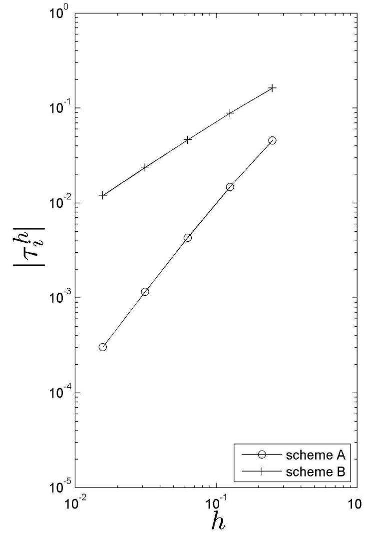
6) The FD formula of Question 3 and δ^+_x are now used to compute the first derivative of $u(x) = \log x$ at grid point $x = 0.8$ on a series of meshes where Δx is halved successively. For both schemes, Figure A.1 shows a loglog plot of the truncation error versus Δx and the corresponding datas are reported in the table. Calculate the missing values of R in the table and deduce what are Scheme A and B.

Exercice A.2 (FD formulas on non-uniform mesh).

On the non-uniform mesh $\{x_i\}$ with grid size $\Delta x_i = x_i - x_{i-1}$, we consider the following centered FD formula to compute $\left(\frac{\partial u}{\partial x}\right)_i$:

$$\delta^0_x u_i = \frac{1}{\Delta x_{i+1} + \Delta x_i}\left[\frac{\Delta x_i}{\Delta x_{i+1}}(u_{i+1} - u_i) + \frac{\Delta x_{i+1}}{\Delta x_i}(u_i - u_{i-1})\right] \tag{A.3}$$

1) Show that this formula leads to the usual centered formula on a uniform mesh.
2) Show that this formula is second-order accurate on non-uniform mesh.



Δx	Scheme A, $\lvert\tau_i^{\Delta x}\rvert$	R	Scheme B, $\lvert\tau_i^{\Delta x}\rvert$	R
0.25	4.55×10^{-2}	–	1.62×10^{-1}	–
0.25/2	1.48×10^{-2}		8.85×10^{-2}	
0.25/4	4.31×10^{-3}		4.64×10^{-2}	
0.25/8	1.17×10^{-3}		2.38×10^{-2}	
0.25/16	3.04×10^{-4}		1.21×10^{-2}	

Fig. A.1. At left : loglog plot of the truncation error versus grid-size, at right : corresponding datas.

Exercice A.3 (FDM for elliptic equations and conditioning of FD matrices).

Consider the Laplace equation :

$$-u''(x) = f(x) \quad \text{dans } \Omega =]0,1[, \tag{A.4}$$

with homogeneous Dirichlet boundary conditions. We define the uniform mesh $\Omega_i = \{x_i = i\Delta x,\ i = 0, \ldots, N+1\}$ with grid-size $\Delta x = 1/(N+1)$.

1) Show that solving (A.4) by centered FD scheme leads to the following linear system :

$$\frac{1}{\Delta x^2}\mathcal{K}U = F \tag{A.5}$$

where U and F denote respectively the column vector of unknowns $\{u_i\}$ and the RHS $\{f(x_i)\}$. The $N \times N$ tridiagonal matrix $\mathcal{K} = \text{Tridiag}_N[-1, 2, -1]$ reads :

$$\mathcal{K} = \begin{bmatrix} 2 & -1 & & & \\ -1 & 2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2 & -1 \\ & & & -1 & 2 \end{bmatrix}.$$

2) Show that eigenvalues of $\mathcal{K}$ can be expressed[1] as :

$$\lambda_k = 2 - 2\cos k\pi\Delta x, \qquad \text{for} \quad k = 1, \ldots, N, \tag{A.6}$$

Enumerate the main properties of $\mathcal{K}$.

[1] For a matrix of the form $\text{Tridiag}_N[a, b, c]$, eigenvalues read :

$$\lambda_k = b - 2\sqrt{ac}\cos\frac{k\pi}{N+1}, \qquad k = 1, 2, \ldots, N.$$

3) By noticing that :

$$\cos N\pi\Delta x = -\cos\pi\Delta x,$$

prove that the conditon number of $\mathcal{K}$ can be written

$$\chi(\mathcal{K}) = \frac{1+\cos\pi\Delta x}{1-\cos\pi\Delta x},$$

and when $\Delta x \to 0$,

$$\chi(\mathcal{K}) \simeq \frac{4}{\pi^2\Delta x^2}.$$

4) What would be the performances of an iterative method for solving this linear system ?

A.2 FD Methods II. Analysis of FD schemes for unsteady problems: consistency, stability.

Exercice A.4 (The θ-scheme for parabolic equations).

This exercise concerns FD approximation of the 1D heat equation :

$$\frac{\partial u}{\partial t} - \nu\frac{\partial^2 u}{\partial x^2} = f(x,t), \tag{A.8}$$

with $\nu \geq 0$. Temporal and spatial steps are respectively denoted by Δt and Δx, u_i^n is an approximation of $u(t_n, x_i)$ at time $t_n = n\Delta t$ at location $x_i = i\Delta x$. Spatial discretization uses the FD centered scheme and time-stepping is based on the θ-scheme which reads for the ODE $\dot{y}(t) = f(y(t), t)$:

$$\frac{y^{n+1}-y^n}{\Delta t} = \theta f(t_{n+1}, y^{n+1}) + (1-\theta) f(t_n, y^n), \tag{A.9}$$

where θ is a real parameter in $[0,1]$. Boundary conditions are discarded.

1) Use the method of lines (MOL) to discretize the heat equation with centered FD formulas. Give the corresponding system of ODEs in both matrix and index forms (use the δ^0_{xx} notation).
2) Complete the discretization using the θ-scheme. How many time-levels possess the scheme ? For which value of θ is the scheme implicit, explicit ? What common FD schemes correspond to $\theta = 0$ and $\theta = 1$.
3) Approximate $\theta\phi^{n+1} + (1-\theta)\phi^n$ at time $n\Delta t$ by using Taylor expansion (up to order 2).
4) Give the two first terms of the truncation error of $(u^{n+1} - u^n)/\Delta t$ and $\delta^0_{xx} u_i$ which are respectively used to discretize $\partial u/\partial t$ and $\partial^2 u/\partial x^2$.
5) Deduce from the previous questions the the truncation error of the θ-scheme at $(n\Delta t, i\Delta x)$, up to second order terms $\Delta t^2, \Delta x^2$.
6) Study the consistency of the θ-scheme with respect to the values of θ.
7) The scheme built in Question 2) with $\theta = 1/2$ is the so-called *Crank-Nicolson scheme* (CN). Write the CN scheme in its *iterative* form :
$$\sum_{j=i-1}^{i+1} \mathcal{B}^1_{ij} u_i^{n+1} = \sum_{j=i-1}^{i+1} \mathcal{B}^2_{ij}\, u_j^n,$$
where $\mathcal{B}^l$ denote the *iteration matrices* of the scheme. Express the matrix coefficients as a function of the diffusion number $r = \nu\Delta t/\Delta x^2$.
8) Express the amplification factor $g(\xi)$ of the CN scheme as a function of r and $\xi = \sin^2\frac{k\Delta x}{2} = \frac{1-\cos k\Delta x}{2}$. Recall the von Neumann stability condition and study the stability of the scheme.
9) (Optional) Same question in the case of an arbitrary value of θ. Show that if $\theta \geq 1/2$, the resulting scheme is unconditionally stable, and if $\theta < 1/2$, the stability condition reads
$$\Delta t \leq \frac{\Delta x^2}{2\nu(1-2\theta)}.$$

10) Summarize the properties $\theta-$scheme for $\theta = 0$, 1 and 1/2.

Exercice A.5 (The Richardson scheme for the heat equation).

We wish now to discretize the 1D heat equation (A.8) with central FD differences and the time-stepping method known as the *midpoint rule*, which reads for ODE $\dot{y}(t) = f(y(t), t)$:

$$\frac{y^{n+1}-y^{n-1}}{2\Delta t} = f(t_n, y^n), \tag{A.10}$$

1) Discretize the heat equation with the method of lines (MOL). The resulting scheme is known as the Richardson scheme for the heat equation.
2) How many time-levels possess the scheme ? Calculate its truncation error.
3) Perform the von Neumann stability analysis to calculate its amplification matrix (hint : introduce the auxiliary variable $v_i^{n+1} = u_i^n$ to reduce the scheme to a two-level scheme.). Is the Richardson scheme stable ?

A.3 FD methods III : steady equations

Exercice A.6 (Steady advection-diffusion problem).

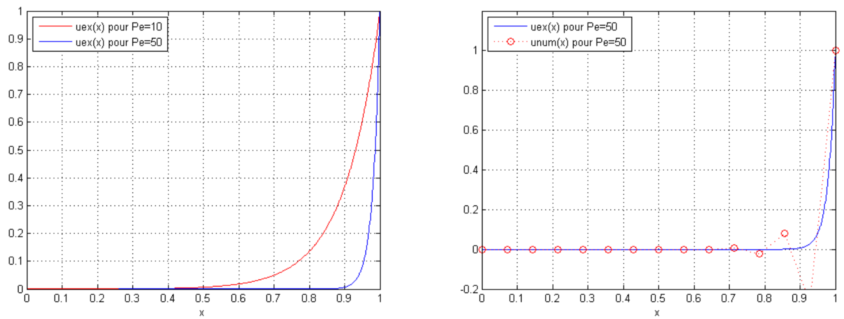


Fig. A.2. At left : Exact solution steady advection-diffusion problem for various Péclet number values. At right : Discrete solution obtained with FD centered scheme $n = 13$.

We consider the following advection-diffusion problem with Dirichlet conditions :

$$au'(x) - \nu u''(x) = 0 \quad \text{in } \Omega =]0,1[, \tag{A.1a}$$
$$u(0) = 0, \quad u(1) = 1, \tag{A.1b}$$

where $\nu > 0$ denotes the viscosity and the convective velocity a is assumed to be strictly positive. The analytic solution of this boundary value problem is :

$$u(x) = \frac{\exp\frac{a}{\nu}x - 1}{\exp\frac{a}{\nu} - 1}. \tag{A.2}$$

1) Give the dimensionless form of (A.1) for which the characteristic length is the domain length L, and a is the characteristic velocity. We denote by $\mathrm{Pe} = aL/\nu$ the Péclet dimensionless number.
2) Show graphically that the exact solution exhibits a boundary layer when convection dominates, *i.e.* $\mathrm{Pe} \gg 1$.

3) Discretize (A.1) with centered FD on mesh $\{x_i = i\Delta x,\ i = 0, \ldots, N+1\}$ where $\Delta x = 1/(N+1)$ is the grid size. Write the corresponding discrete system in matrix form:

$$\sum_{j=i-1}^{i+1} A_{i,j} u_j = RHS_i, \quad \text{for all } i = 1, \ldots, N. \tag{A.3}$$

and gives the coefficients of the scheme as a function of the *cell Péclet number* Pe_h defined by :

$$\mathrm{Pe}_h = \frac{a\Delta x}{\nu}. \tag{A.4}$$

What is the order of accuracy of the scheme ?

4) Seek a solution of the central scheme as the *normal mode* $u_i = q^i$. Determine q as a function of the scheme coefficients, and show that the solution reads:

$$u_i = \frac{\left(\dfrac{2+\mathrm{Pe}_h}{2-\mathrm{Pe}_h}\right)^i - 1}{\left(\dfrac{2+\mathrm{Pe}_h}{2-\mathrm{Pe}_h}\right)^{N+1} - 1}, \tag{A.5}$$

Show that discrete solution displays spurious oscillations for $\mathrm{Pe}_h > 2$.

5) Numerical application : consider (A.1) with $\nu = 10^{-3}$ and $a = 1$. Calculate the value of the Péclet number. Give the minimum number of nodes N for which the solution is free of oscillations.

6) Discretize the convective term in (A.1) with upwind scheme instead of the centered scheme. Show that the discrete solution now reads :

$$u_i = \frac{(1+\mathrm{Pe}_h)^i - 1}{(1+\mathrm{Pe}_h)^{N+1} - 1}, \tag{A.6}$$

and that it is free of oscillations for any value of Pe_h.

A.4 FD method IV : schemes for hyperbolic equations

Exercice A.7 (Analysis of the numerical dissipation and dispersion errors of FD schemes).

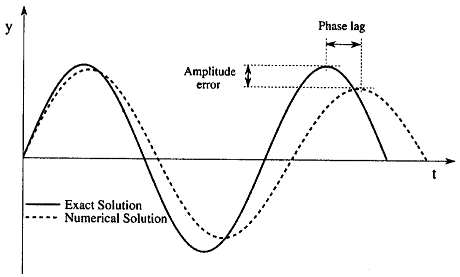


Fig. A.3. The numerical errors of a FD scheme can be decomposed as the sum of *dissipative* errors (alteration of the amplitude of the solution) and *dispersive* or phase lag errors (alteration of the propagation speed of the solution). Dissipative errors are associated with even-order derivatives of the modified equation, while dispersive errors come from odd-order terms.

We consider the hyperbolic transport equation:

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0, \tag{A.7}$$

with $a > 0$ is the convection velocity. The lecture showed that the upwind scheme:

$$\frac{u_i^{n+1} - u_i^n}{\Delta t} + a\,\delta_x^- u_i^n = 0, \tag{A.8}$$

corresponds to a stabilization of the explicit central scheme (unconditionally unstable) by adding a term of numerical dissipation $-\nu_{\mathrm{num}}\delta_{xx}^0$, where $\nu_{\mathrm{num}} = a\Delta x/2 > 0$ is the artificial viscosity. In the lecture, this dissipation term has been derived heuristically, but now this exercise aims to present rigorously the concepts of numerical dissipation (and numerical dispersion) of a FD scheme. For this purpose, we now adopt an approach based on the Fourier analysis of the *equivalent differential equation* or *modified equation* of FD schemes. This analysis will be illustrated in this exercise for the case of the upwind scheme (A.8), but it can be applied to any FD scheme and other types of PDEs.

Let us denote by

$$A\,u(t,x) = 0, \tag{A.9}$$

the PDE we wish to solve (here it is Eq. (A.7)), and by

$$A_h\,u_i^n = 0, \tag{A.10}$$

the FD scheme (here, Eq. (A.8)). Then, by using Taylor expansions one can write that the exact solution $u(t,x)$ verifies at each grid point:

$$A_h\,u(t_n,x_i) = A\,u(t_n,x_i) + \tau\big(u(t_n,x_i)\big), \tag{A.11}$$

where $\tau\big(u(t_n,x_i)\big)$, or simply τ_i^n, is the LTE of the scheme which involves the scheme parameters Δt and Δx and a series of temporal and spatial derivatives of $u(t,x)$. Examining the leading term in this series gives the truncation error of the scheme, *i.e.* $\tau_i^n = \mathcal{O}(\Delta t, \Delta x)$ for the upwind scheme.

The modified equation analysis is a generalization of this truncation error analysis: according to (A.11), the discrete solution of the FD scheme verifies the *equivalent differential equation*:

$$A\,u(t,x) + \tau\big(u(t,x)\big) = 0, \tag{A.12}$$

where $\tau(u)$ involves only spatial derivatives of the solution[2]. Since $\tau(u)$ involves an infinity of terms, Eq. (A.12) is of little practical use. However, a Fourier analysis of this equation (similar to the analysis performed in the first lecture on PDEs) will give us information on the error properties of the FD scheme, and notably its numerical dissipation and dispersion errors, see Fig. A.3.

The principles of the modified equation analysis are now illustrated for the case of the upwind scheme (A.8) by performing the following steps:

1) Calculate the leading terms of the truncation error of the upwind scheme (up to $\mathcal{O}(\Delta t^3, \Delta x^3)$). Show that the LTE is first-order in time and space.

2) Replace the time derivatives of the truncation error $\tau\big(u(t,x)\big)$ by spatial derivatives and show that the *equivalent differential equation* (A.12) of the upwind scheme is :

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} + \eta\frac{\partial^2 u}{\partial x^2} + \Delta\frac{\partial^3 u}{\partial x^3} + \mathcal{O}(\Delta x^3) = 0, \tag{A.13}$$

where the parameters η and Δ depend on a, Δx and C the Courant number[3].

3) According to the lecture, the solution of the transport equation (A.7) sought as the Fourier mode $u(t,x) = \widehat{u}(t)\,e^{ikx}$ reads:

$$u(t,x) = \widehat{u}_0\,e^{ik(x-at)}, \quad \text{where } \widehat{u}_0 \text{ depends on the initial condition,} \tag{A.14}$$

showing that the exact solution is a wave propagating at constant speed $a > 0$ with no deformation of its amplitude. A similar analysis of the wave computed by the upwind scheme is now performed by considering the equivalent equation (A.13) reduced to its lower order terms, *i.e.* :

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} + \eta\frac{\partial^2 u}{\partial x^2} + \Delta\frac{\partial^3 u}{\partial x^3} = 0. \tag{A.15}$$

---

[2] The time derivatives in $\tau(u)$ are replaced by spatial derivatives by using the original PDE (A.9) as in the lecture on the Lax-Wendroff scheme.

[3] For the case $C = 1$, you will notice that the truncation error exactly cancels out, which means that the FD solution is free of any numerical error. This is a particular case that occurs only for 1D linear PDEs when $C = 1$, and never for more general PDEs, which are usually nonlinear and multidimensional with different convection speed for each dimensions. Hence we will never set $C = 1$ in the Matlab exercises, and rather use $C \simeq 1$ in order to visualise the effects of dispersion and dissipation errors.

Prove that the Fourier solution of this equation can be written as :

$$u(t,x) = u_0'(t)\, e^{ik(x-a't)}, \quad \text{with } a' \in \mathbb{R}, \tag{A.16}$$

and express $u_0'(t)$ as a function of $\widehat{u}_0$, η and k, and a' as a function of a, Δ and k.

4) What happens if $\eta > 0$? Compare with the *CFL* condition of the upwind scheme.

5) Plot the exact (A.14) and numerical (A.16) solutions to the transport equation in the same figure as in Fig. A.3. Identify the terms in (A.15) which are responsible of numerical dissipation and those responsible of numerical dispersion. Deduce the order of the dissipation and dispersion error of the upwind scheme[4] .

[4] In addition, it can be proved that the modified equation of the Lax-Wendroff scheme applied to the transport equation reads:

$$\frac{\partial u}{\partial t} + a\,\frac{\partial u}{\partial x} + a\frac{\Delta x^2}{6}(1-C^2)\frac{\partial^3 u}{\partial x^3} + a\,C\frac{\Delta x^3}{8}(1-C^2)\frac{\partial^4 u}{\partial x^4} + \mathcal{O}(\Delta t^4, \Delta x^4) = 0,$$

which shows that the LW scheme is second-order in time and space, its dispersive error is second-order and its dissipative error is third-order. The examination of the sign of the first dissipative term's coefficient in the truncation error yields the CFL stability condition of the scheme.

B Finite-volume method (FVM) for CFD

B.1 FVM on structured meshes

Exercice B.1. (FVM for 1D convection-diffusion equation)

We consider the steady convection-diffusion equation

$$\frac{\mathrm{d}u\phi}{\mathrm{d}x} - \frac{\mathrm{d}}{\mathrm{d}x}\left(\nu\frac{\mathrm{d}\phi}{\mathrm{d}x}\right) = q(x), \tag{B.1}$$

in the spatial domain $\Omega =]0,1[$, where the right-hand-side $q(x)$ is a given data. The viscosity $\nu = \nu(x)$ is supposed strictly positive for all $x \in \Omega$. The convection velocity $u = u(x)$ has arbitrary sign, except at the boundaries where $u(x=0) > 0$ and $u(x=1) > 0$: this means that at boundary $x = 0$ the fluid enters the domain, while at $x = 1$ the fluid exits the domain. To ensure that the mathematical problem is well posed, we chose the following boundary conditions : at $x = 0$ we impose the *inflow* condition of Dirichlet type:

$$\phi(0) = g_-, \tag{B.2}$$

and at $x = 1$ an outflow condition of Neumann type :

$$\frac{\mathrm{d}\phi}{\mathrm{d}x}(1) = g_+. \tag{B.3}$$

The purpose of the exercise is to write down the FV discretization of this boundary value problem on a non-uniform cell-centred mesh.

1) *Mesh construction.* Let $\{x_{i+\frac{1}{2}},\ i = 0,\dots,N\}$ a series of nodes such as :

$$0 = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N+\frac{1}{2}} = 1.$$

Build a cell-centred mesh such as $x_{i+\frac{1}{2}}$ represented the boundaries of cell Ω_i, whose center is x_i. The volume of cell Ω_i is Δx_i, and the spacing between the nodes x_i and x_{i+1} is denoted $\widetilde{\Delta x}_i$. How many cells are present in the mesh? Make a sketch of the mesh with precise notations for a typical cell away from the boundaries, and for cells Ω_1 and Ω_N adjacent to the boundaries $x = 0$ and $x = 1$.

2) *Specification of the discrete unknowns.* We define cell-averages as:

$$\phi_i = \frac{1}{\Delta x_i}\int_{\Omega_i}\phi(x)\,\mathrm{d}x, \qquad q_i = \frac{1}{\Delta x_i}\int_{\Omega_i} q(x)\,\mathrm{d}x, \qquad i = 1,\dots,N.$$

Discretize these volume integrals in order to link cell-average values to nodal values on the mesh.

3) *Finite-volume formulation.* Integrate Eq. (B.1) on the control volume Ω_i, and shows that the FV discretization reads:

$$F_{i+\frac{1}{2}} - F_{i-\frac{1}{2}} = \Delta x_i\, q_i, \quad i = 1,\dots,N, \tag{B.4}$$

where $F = F^{\mathrm{c}} - F^{\mathrm{d}}$ is the discret flux, and write down the expression of the conservative and dissipative flux fonctions. What is the main difference with the multidimensional case of the lecture notes?

4) *Flux interpolation.* The discrete fluxes $F_{i+\frac{1}{2}}$, $F_{i-\frac{1}{2}}$ needs now to be interpolated with respct to the nodal values ϕ_i.

   a) Perform the interpolation of the diffusive flux $F^{\mathrm{d}}_{i+\frac{1}{2}}$ for $i = 1,\dots,N-1$. On uniform mesh with $\nu(x) = \nu > 0$, show that the usual finite-difference formula is recovered :

$$\frac{F^{\mathrm{d}}_{i+\frac{1}{2}} - F^{\mathrm{d}}_{i-\frac{1}{2}}}{\Delta x} = \nu\delta^0_{xx}\phi_i.$$

b) In the case of a convection velocity of arbitrary sign, write down the UDS interpolation of the convective flux $F^{\mathrm{c}}_{i+\frac{1}{2}}$ for $i = 1, \dots, N-1$, *i.e.* away from the boundaries. Show that the upwind finite-difference formula δ_x^- is recovered when $u(x) = u > 0$.

c) Same question for the CDS interpolation.

5) *Boundary conditions implementation.* The interpolation of the discrete fluxes near the boundaries needs to take into account the boundary conditions.

a) According to Eq. (B.4) and the sketch of the mesh, which are the discrete fluxes whose interpolation needs to be modified according to the boundary conditions?

b) Implement the boundary conditions in the diffusive fluxes.

c) For the UDS interpolation, implement the boundary conditions in the convective fluxes. Is the outflow condition necesary?

d) Same question for the CDS interpolation.

6) *Matrix form of the scheme.* Show that the FV discretizaton leads to the $N \times N$ linear system :

$$(\,\mathcal{C}_h(u) - \mathcal{D}_h\,)\,\Phi = Q'.$$

For the UDS interpolation, give the expresson of the right-hand-side Q' and the tridiagonal matrices $\mathcal{C}_h(u)$ and $\mathcal{D}_h$.

Exercice B.2 (The *QUICK* and κ-scheme for convection discretization).

This exercise concerns the construction of higher-order upwind-biased finite-volume scheme for the convective fluxes. We will use the standard notations of the lecture notes for the 2D convection diffusion equation: the convective flux $F_x^{\mathrm{c}} = u\phi$ is discretized on the mesh depicted in Fig 15.2 using mid-point rule as $(F_x^{\mathrm{c}})_{\mathrm{e}} = u_{\mathrm{e}}\, \Delta y_{\mathrm{P}}\, \phi_{\mathrm{e}}$ at face Γ_{e}. The CDS and upwind interpolation of ϕ_{e} have been described in the lectures notes, and in the following we will build high-order interpolation for ϕ_{e}. Throughout this exercise, we will suppose that $u_{\mathrm{e}} > 0$ and that the mesh is uniform of size Δx.

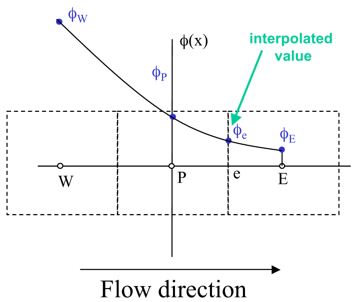


Fig. B.1. Parabolic interpolation used by the *QUICK* scheme.

1) While the CDS scheme uses linear polynomial interpolation between ϕ_{E} and ϕ_{P} to build a second-order formula, the *QUICK* scheme[1] uses quadratic interpolation between ϕ_{E}, ϕ_{P} and ϕ_{W} to achieve better accuracy. Therefore, as the UDS scheme, the *QUICK* scheme uses upwind-biased interpolation following the flow direction (sign of convective velocity u_{e}).

a) Build[2] the quadratic polynomial $\pi_2[\phi](x)$ that fits ϕ at values ϕ_{W}, ϕ_{P} and ϕ_{P} (see Fig. B.1).

[1] Acronym for "Quadratic Upwind Interpolation for Convection Kinematics", introduced by Leonard (1979).

[2] Hint: use the Lagrange interpolation polynomial $\pi_2[\phi](x) = \sum_{j=0}^{2} \phi(x_j) L_j(x)$ at points $x_0 \equiv x_{\mathrm{W}} = x_{\mathrm{P}} - \Delta x$, $x_1 \equiv x_{\mathrm{P}}$, $x_2 \equiv x_{\mathrm{E}} = x_{\mathrm{P}} + \Delta x$, where the jth Lagrange polynomial is :

$$L_j(x) = \prod_{\substack{i=0\\ j \neq i}}^{2} \frac{x - x_i}{x_j - x_i}.$$

b) Evaluate this interpolation polynomial at face $x_{\mathrm{e}} = x_{\mathrm{P}} + \Delta x/2$ to obtain the *QUICK* interpolation formula:

$$\phi_{\mathrm{e}} = \frac{3}{8}\phi_{i+1} + \frac{3}{4}\phi_i - \frac{1}{8}\phi_{i-1}, \tag{B.1}$$

where $\phi_{i+1} = \phi_{\mathrm{E}}$, $\phi_i = \phi_{\mathrm{P}}$, and $\phi_{i-1} = \phi_{\mathrm{W}}$.

c) Write the *QUICK* interpolation scheme as:

$$\phi_{\mathrm{e}} = \frac{\phi_{i+1} + \phi_i}{2} - \frac{\Delta x^2}{8}\delta^0_{xx}\phi_i, \tag{B.2}$$

and give an interpretation of this scheme.

d) Determine the truncation error of interpolation (B.1) at face x_{e}. Can you guess the order of accuracy of the *QUICK* scheme for convection?

e) What nodes would the *QUICK* scheme use in the case $u_{\mathrm{e}} < 0$?

2) A generalization of the *QUICK* scheme (B.2) is given by the family of upwind-biased κ−scheme that reads on uniform mesh :

$$\phi_{\mathrm{e}} = \begin{cases} \dfrac{\phi_{i+1} + \phi_i}{2} + \dfrac{\kappa - 1}{4}\Delta x^2 \delta^0_{xx}\phi_i & \text{if } u_{\mathrm{e}} \geq 0, \\ \dfrac{\phi_{i+1} + \phi_i}{2} + \dfrac{\kappa - 1}{4}\Delta x^2 \delta^0_{xx}\phi_{i+1} & \text{if } u_{\mathrm{e}} < 0, \end{cases} \tag{B.3}$$

where κ is a real parameter, such as special values of this parameter correspond to well-known convection discretization:

- $\kappa = 1$ yields the CDS scheme,
- $\kappa = -1$ yields the 2$^{\mathrm{nd}}$ order upwind scheme,
- $\kappa = 1/3$ yields the 3$^{\mathrm{rd}}$ order upwind scheme,
- $\kappa = 1/2$ yields the *QUICK* scheme.

a) Assume that $u_{\mathrm{e}} = u_{\mathrm{w}} = u > 0$ and write down the κ−scheme as a finite-difference quotient:

$$uD_x^{(\kappa)}\phi_i = \frac{u\phi_{\mathrm{e}} - u\phi_{\mathrm{w}}}{\Delta x}, \tag{B.4}$$

and determine the coefficients of the scheme with respect to κ. Show that the special cases $\kappa = 1$ and $\kappa = -1$ yield the CDS and 2$^{\mathrm{nd}}$ order upwind scheme respectively:

$$\delta^0_x\phi_i = \frac{\phi_{i+1} - \phi_{i-1}}{2\Delta x}, \qquad D_x^-\phi_i = \frac{3\phi_i - 4\phi_{i-1} + \phi_{i-2}}{2\Delta x}. \tag{B.5}$$

3) Calculate the truncation error of the κ−scheme and show that it is 3$^{\mathrm{rd}}$ order accurate for $\kappa = 1/3$ and 2$^{\mathrm{nd}}$ order for all other values of κ.

Exercice B.3 (On conservative and non-conservative discretization methods).

The transport equation for a conserved scalar ϕ in a steady incompressible 2D flow (such as $\nabla \cdot \boldsymbol{v} = 0$) reads in conservative form :

$$\nabla \cdot \boldsymbol{v}\phi = 0. \tag{B.1}$$

This equation is to be solved in a rectangular domain Ω such as the boundary conditions are: inflow condition at west boundary, outflow at east boundary, wall ($\phi = 0$) at north and south boundaries.

1) Show that Eq. (B.1) is analytically equivalent to the convective (or nonconservative) form of the transport equation :

$$\boldsymbol{v} \cdot \nabla\phi = 0. \tag{B.2}$$

2) Integrate Eq. (B.1) in the whole fluid domain and express the compatibility condition verified by ϕ at inflow and outflow boundaries.

3) Discretize Eq. (B.1) with FVM on a Cartesian grid with uniform cells of size Δx and Δy. Interpolation at cell centers is not needed: the values at cell faces will be denoted as $\phi_{i+1/2,j}$, $u_{i+1/2,j}$ on east face, etc ....
4) Integrate the FV scheme of question 3) in the whole fluid domain Ω and compare the result with the analytical compatibility condition of question 2).
5) Discretize Eq. (B.2) with FDM using same notations as question 3). Compare the FD and FV schemes and show that they are equivalent for constant convection velocity only.
6) Same question as 4) for the FD scheme. Show that the compatibility condition is asymptotically verified when the mesh is refined.

B.2 FVM on unstructured meshes

Exercice B.4 (Gradient approximation at cell faces and spurious oscillations).

In this exercise, we will apply the various unstructured gradient approximations of the lecture notes to the particular case of a (orthogonal) rectangular cell of size Δx and Δy. The notations used are the compass notations of Fig. 15.2 of the lecture notes.

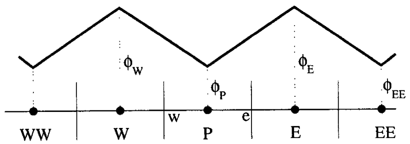


Fig. B.2. Example of spurious oscillations occurring in cell-based gradient approximation.

1) Calculate the face area vector $\boldsymbol{\Delta S}_\mathrm{f}$ for each face e, w, n and s of the cell. Check that they sum up to zero.
2) Calculate the volume of the cell by using Gauss formula:

$$V_\mathrm{P} = \frac{1}{2} \sum_{\text{faces}} \boldsymbol{x}_\mathrm{f} \cdot \boldsymbol{\Delta S}_\mathrm{f},$$

where $\boldsymbol{x}_\mathrm{f}$ is any position vector on face f.
3) Calculate the cell-average gradient of ϕ_P using Green-Gauss cell-based method. Compare with the usual FD central approximation $\delta_0^x \phi_\mathrm{P}$.
4) Calculate the derivative $\left(\overline{\partial\phi/\partial x}\right)_\mathrm{e}$ at face e using arithmetic interpolation. Show that this formula gives rise to checkerboard oscillations as shown in Fig. B.2 by calculating the derivative of $\phi_i = \text{Const.} + (-1)^i$. Compare with the formula obtained with the minimum correction approach.

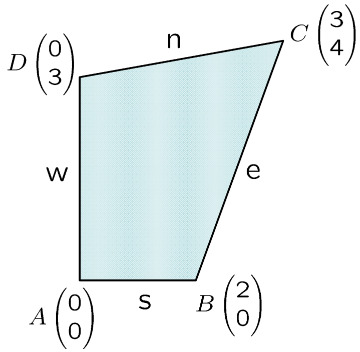
Exercice B.5 (Steady incompressible flow in a general quadrilateral cell).

We consider a steady incompressible 2D flow over the quadrilateral cell Ω depicted in Fig. B.3.

1) Calculate the face area vector $\boldsymbol{\Delta S}_\mathrm{f}$ for each face of the cell. Check that they sum up to zero and complete the table in Fig. B.3.
2) Calculate the mass flow

$$\dot{m}_\mathrm{f} = \int_{\Gamma_\mathrm{f}} \boldsymbol{v} \cdot \boldsymbol{n} \, \mathrm{d}S \cong \boldsymbol{v}_\mathrm{f} \cdot \boldsymbol{\Delta S}_\mathrm{f},$$

on faces n, w and s. Complete the table in Fig. B.3.
3) Use the continuity equation to find the x-velocity component on the east face.
4) Calculate the cell volume with Gauss formula.
5) Calculate the cell-averaged gradient of u and v. Deduce the cell-averaged values of the vorticity $\omega = \partial v/\partial x - \partial u/\partial y$ and the divergence $\partial u/\partial x + \partial v/\partial y$. Why is this last quantity equal to zero?



Face	u_f	v_f	ΔS^x_f	ΔS^y_f	$\dot{m}_\mathrm{f}$
e		0			
n	6	3			
w	5	2			
s	5	2			

Fig. B.3. Quadrilateral cell Ω with vertices A, B, C and D. The velocity value $\boldsymbol{v}_\mathrm{f} = (u_\mathrm{f}, v_\mathrm{f})$ at each face centroid is given in the table.

6) For 2D incompressible flows, the streamfunction ψ is defined in Cartesian coordinates as :

$$u = -\frac{\partial\psi}{\partial y}, \quad v = \frac{\partial\psi}{\partial x}.$$

Note that ψ is defined up to an arbitrary constant, and that opposite sign convention is also valid. A generalization of this definition is :

$$\psi_A - \psi_B = \int_A^B \boldsymbol{v} \cdot \boldsymbol{n} \, \mathrm{d}S, \tag{B.1}$$

which states that the change in ψ between points A and B is equal to the flow rate between any curve Γ connecting A to B.
a) Assuming $\psi_A = 0$, use Eq. (B.1) to calculate the streamfunction values at the other vertices of the cell.
b) Calculate the streamfunction at vertex A by using the value of ψ_D you found at the previous question. Compare with the initial value $\psi_A = 0$.

Lab Exercice on Computational Fluid Dynamics

First steps with ANSYS FLUENT: The axysimmetric expansion flow

Instructor: Olivier Botella `olivier.botella@univ-lorraine.fr`

1 Problem statement

The purpose of this lab exercise[1] is to learn the basics of CFD analysis with the commercial software ANSYS FLUENT[2] by computing the axisymmetric expansion flow presented in the lectures (Fig. 1).

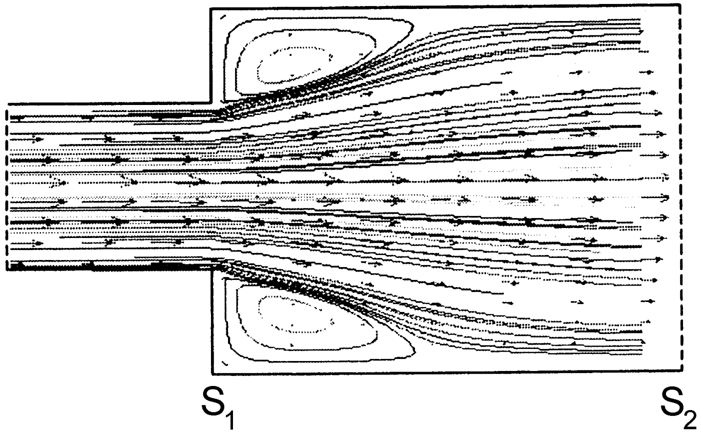


Figure 1: Velocity vectors at the vicinity of the sudden expansion (section S_1 at $x_1 = 1.0\,m$). In the upstream pipe the flow is fully developped, then the singularity causes abrupt flow separation, which causes recirculation zones. In the downstream pipe, the flow reattaches at Section S_2 and reaches fully developed regime.

This analysis will be performed with the ANSYS WorkBench platform, which integrates the following softwares:

- Geometry creation (DesignModeler)
- Mesh generation (Ansys Meshing)
- Flow setup and numerical solution (Fluent)

[1]The pdf version of this document and all necessary documents for completing the exercise are available online at the course webpage: *UE 203 - Computational Fluid Dynamics M1 MEPP* at the ARCHE website (`https://arche.univ-lorraine.fr/course/view.php?id=9396`).

[2]This lab exercise has been performed with Version 15.0 of ANSYS WorkBench , and revised with V2025 R1.

Work to be performed

This document will introduce you to the basic functionality of the various WorkBench softwares and will guide you through the CFD analysis of the expansion flow.

Read carefully the document and execute with the WorkBench the tasks presented in the document. You should answer to the various questions of the document in a Word report. You will send me your report at the end of the CFD lectures as part of your evaluation.

2 The WorkBench platform

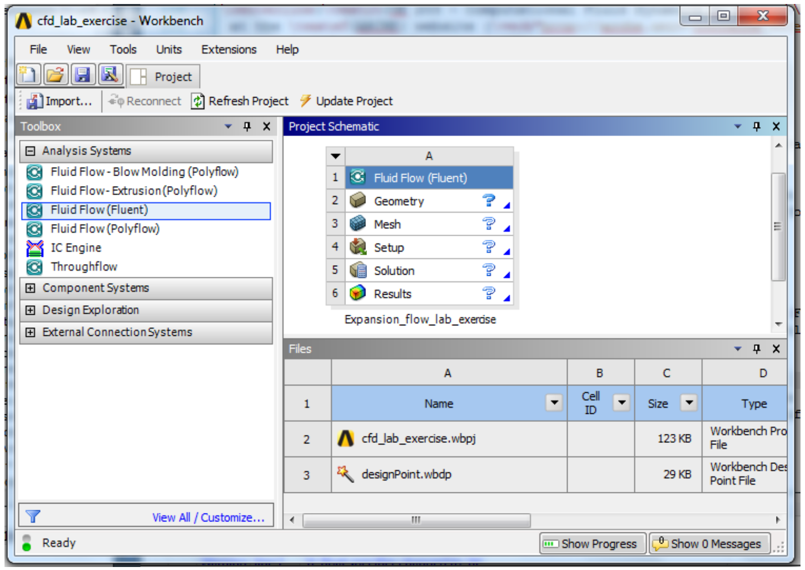


Figure 2: Graphical user interface (GUI) of ANSYS WorkBench .

Launch the ANSYS WorkBench software from your PC and create a Fluid Flow (Fluent) project. The Fluent workflow shown in Fig. 2 shows the consecutive steps of the CFD analysis. WorkBench will automatically manage the various data files generated by your Fluent project.

Save now the WorkBench project (**.wbpj* file) on your disk with a meaningful and correct names for your file and working directory (no special characters, foreign accents, avoid spacings), or the software may malfunction and your data be lost!

The language preferences of the various WorkBench softwares can be modified with the option Regional and Language Options from the menu bar: Tools → Options... You need to restart WorkBench to make the change effective.

3 Computational domain creation with DesignModeler

According to the symmetries of the flow domain of Fig. 1, we assume that the flow is asymmetric so that it is only required to create the computational domain shown in Fig. 3. These assumptions allow us to do 2D computations instead of 3D, that will save computational time and memory space.

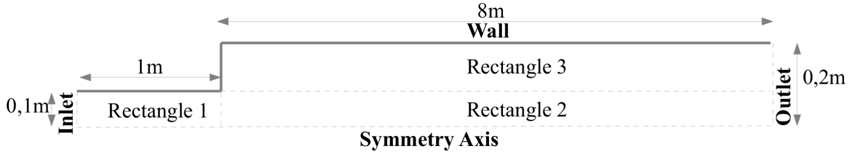


Figure 3: Computational domain and boundary conditions for the cylindrical expansion flow. The 2D computational domain is composed of 3 rectangular surfaces.

For axisymmetric flow computations with ANSYS Fluent , it is important to:

- Create the computational domain in the upper half XY plane (such as $y > 0$) of DesignModeler .
- Ensure that the symmetry axis of the domain is the horizontal axis ($y = 0$).

On WorkBench version 16.0 and above, the default geometry software may be SpaceClaim. To change the default software to DesignModeler , go to the WorkBench menu bar: Tools → Options... Geometry Import.

3.1 Overview of DesignModeler

The graphics interface (GUI) of DesignModeler is shown in Fig. 4. The most relevant components are:

- Graphics Window: sketching area where the various geometrical components of the computational domain are viewed and manipulated with the mouse.
- Menu Bar: where you can read/write the geometry files, create new sketches, *etc.*..
- Toolbars: The most important tools are:
  - View toolbars : icons for moving and zooming the geometry with the mouse in the graphics window.
  - Selection toolbars : Icons for mouse selection of the various elements of the geometry (vertex, edge, face and volume) in the graphics window.
  - New sketch : A "sketch" is a 2D drawing of the contours of the computational domain.
- Mode Tabs: there are 2 different modes:

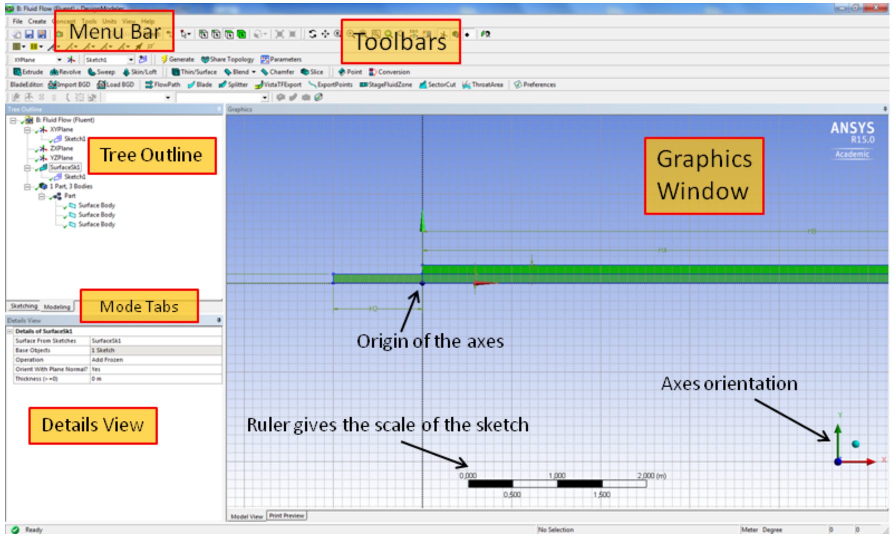


Figure 4: GUI of DesignModeler . Note that 2D geometries should be created in the XY plane. Click on the axes to change orientation of the graphics window.

  - Sketching mode: where you draw the contours of the geometry with the help of the various sketching tools shown in Fig. 5.
  - Modeling mode: that shows the Tree outline, where the basic operations for creating the geometry are shown sequentially: the default planes (where the sketches are stored), the geometry operations on the sketches, the list of created parts.
- Details View: where details of the selected geometry (dimensions, constraints, options, *etc.*..) are displayed and can be modified. Note that grey colored details cannot be modified.

It is important that you memorize the structure of this GUI, since you will use it extensively to do the lab exercises. Note that the GUI of Ansys Meshing is structured similarly.

3.2 Generation of the computational domain

DesignModeler creates the computational domain with the following consecutive steps:

Step 1) Sketching mode: draw the contours of the domain in one (or several) 2D sketches. It is easier to split the computational domain in several elementary entities (like the 3 rectangles of Fig. 3) and draw them separately.

Step 2) Modeling mode: create "solid" bodies (in 2D, it is a surface body) from these contours.

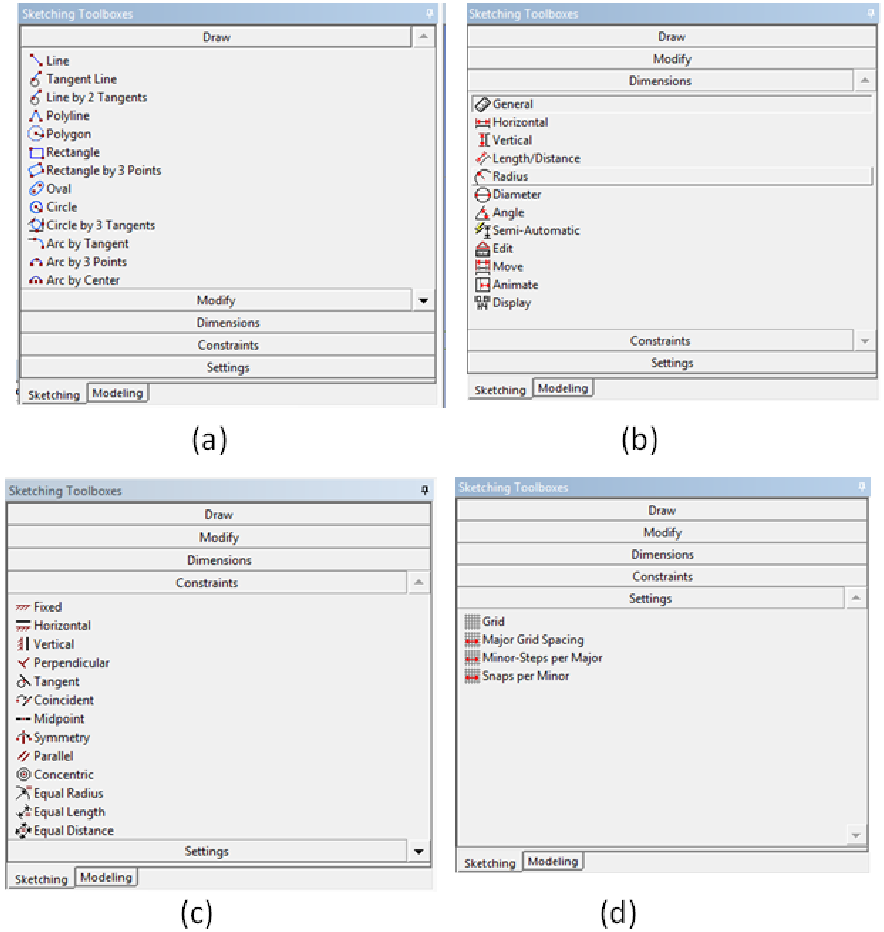


Figure 5: Toolbox available for sketching. (a) Draw: tools for drawing elementary 2D entities (lines, rectangles, circles, ...), (b) Dimensions: tools for setting the dimensions of the entities (by mouse selection in the graphic windows), (c) Constraints: tools for imposing constraints on each entities to assemble the final geometry (alignment with the axes, *etc.*..), (d) Settings: tools for displaying grid lines in the graphic window to make sketching easier.

Step 3) Modeling mode: combine these bodies into a unique computational domain (which is called a part in DesignModeler).

Apply now this strategy to create the domain of Fig. 3. In the same sketch sketch, you will draw each of the 3 rectangles shown in this figure, then assemble them as a unique part with 3 distinct surface bodies.

Note that the Undo/Redo tools are only available in the sketching mode. In the Modeling Mode, you can either Suppress a body[3] or Delete it permanently from the tree outline.

Details View

Details of SurfaceSk1	
Surface From Sketches	SurfaceSk1
Base Objects	1 Sketch
Operation	Add Frozen
Orient With Plane Normal?	Yes
Thickness (>=0)	0 m

Figure 6: Base Objects: the 3 selected rectangles. Add Frozen (in French: Ajouter Corps Bloqué): it means that the three surfaces remain independent in the computational domain. You will be able to mesh each surface independantly in Ansys Meshing . Note that grey colored details cannot be modified.

0) In the Tree Outline of the modeling mode, click on XYPlane and create a new sketch . Go to the Settings toolbox (Fig. 5 (d)) to show the grid and apply adequate parameters for viewing the geometry: Major Grid spacing: 1 m, Minor-steps per Major: 10.

1) In the sketching mode, create Rectangle 1 with the Draw toolbox of Fig. 3 and set its dimensions[4]. Repeat for the other 2 rectangles. Align them as shown in Fig. 4 by using the Constraints toolbox[5].

2) In the Tree Outline of the modeling mode, select the sketch and click on Concept → Surfaces From Sketches from the menu bar. A surface sketch SurfaceSk1 is created in the tree outline, whose details are shown on Fig. 6. To finalize the operation click on Generate .

3) You should have created 3 parts made from the 3 surface bodies you have drawn (look in the Tree Outline). Now, to assemble the 3 bodies in the same part, select them all and right-click on Form New Part. As a result, you have created the computational domain as 1 part made of 3 surfaces bodies, as in the tree outline of Fig. 4.

4) The last step is to define the computational domain as “fluid”. To do this, set Fluid/solid to be Fluid in the Details View of the part.

[3] It will be ignored when the geometry is generated.
[4] Define them with Dimensions toolbox and set their values in the Detail View.
[5] Use the Coincident constraint to stitch the edges of each rectangle to the X or Y axes.

If an element is highlighted with a yellow bolt in WorkBench , it means that the software waits for the user to update/generate this element.

Question 1. *Once the computational domain has been created, display it in your* Word *report and exit* DesignModeler .

When you exit the software, the computational domain is automatically saved on disk in the DesignModeler database format (**.agdb*), and ready to be used by the next software in the WorkBench workflow. You can view all files generated during the lab exercise (and their location on your hard-disk) by toggling View → Files in the WorkBench menu bar.

4 Mesh generation with Ansys Meshing

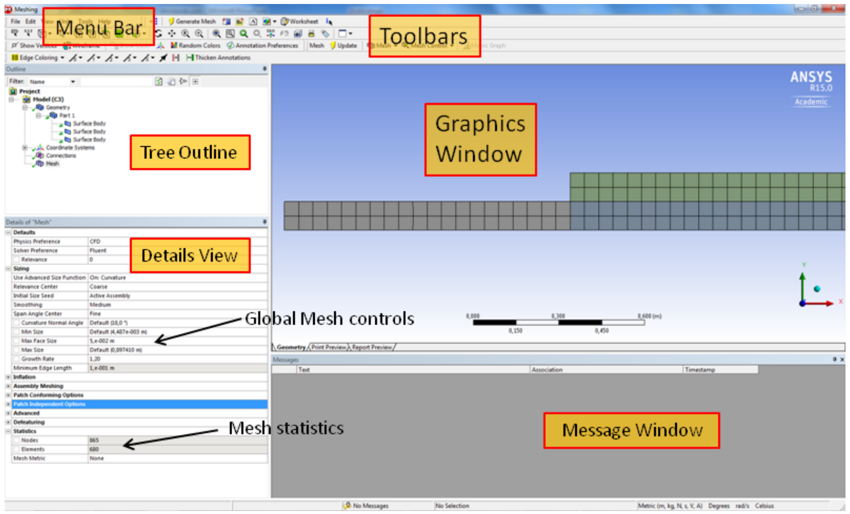


Figure 7: GUI of Ansys Meshing . The quad mesh shown in the graphics window has been generated with Element Size=0.05 m and all other parameters set to default values.

Open now Ansys Meshing from the WorkBench . The GUI, shown in Fig. 7, is similar to DesignModeler with the same mouse functionality. In this first lab exercise, we will only consider the generation of the most simple case of quadrilateral mesh, *i.e.* the Cartesian mesh (see the CFD lectures). Ansys Meshing is able to generate a mesh automatically, with limited user's input. For example, after clicking on the Generate Mesh tool, you should have to create the uniform mesh shown in Fig. 7. The only user input is the (Element Size in the Details View of the mesh).

Question 2. *Try generate meshes with various cell sizes. Display the resulting meshes in your* Word *report with the corresponding number of cells (Elements) and vertices (Nodes) given in the Statistics detail view of the mesh.*

4.1 Generation of a graded Cartesian mesh

The flaws of the "automatic" mesh of Fig. 7 is that the user does not control the mesh resolution in the vicinity of strong flow gradients, which are located in the boundary layers along solid walls and in the recirculation zone (see Fig. 1). Ansys Meshing allows various strategies for refining locally the mesh in selected areas of the computational domain. In this lab exercise, we will use the "bottom up" strategy to create a better quality mesh: first introduce a 1D meshing of the edges of the domain, then use a quad method to mesh the computational domain (which is called a Face, since it is 2D). The resulting mesh, shown in Fig. 8, is generated by the following steps:

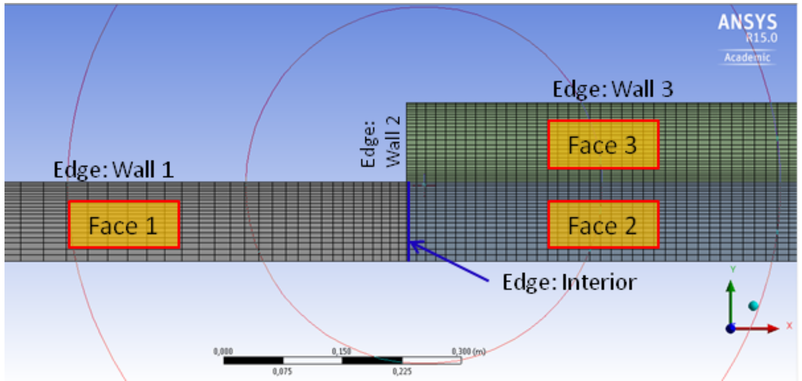


Figure 8: Zoom of the mesh near the singular expansion. Note that the mesh is refined near the walls and the salient corner. This mesh has 14300 cells.

Edge	Interior	Wall 1	Wall 2	Wall 3
# of divisions	20	40	25	300
Bias type	_ _ — ——	_ _ — ——	_ — —— — _	_ _ — ——
Bias factor	3	5	2,5	4

Table 1: Sizing details for the edges shown in Fig. 8.

1) Edge meshing (Tool for edge selection): In the Graphics Window, select one of the edges of Face 1 and insert an edge sizing method[6]. The Edge Sizing method appears in the Tree Outline of the mesh. Choose Type: Number of Divisions and enter the details given in Table 1 (more informations are given at the end of the section). Repeat for the 3 other edges of Face 1.

2) Face meshing (Tool for face selection:) Complete the meshing of Face 1 by inserting a Mapped Face Meshing with Method: Quadrilaterals. Generate the mesh of Face 1 by right-clicking on Generate Mesh on Selected Bodies.

[6]Right-click on the moose: Insert → Sizing.

3) Preview the mesh by clicking on Mesh in the Tree Outline. If the mesh density is not similar to Fig. 8 or the grid lines are not parallel to the axes, you certainly made a mistake in some edge sizing details. In this case, delete the mesh[7], review your edge sizings and generate again the mesh of Face 1.

4) Repeat these steps for Face 2 and 3.

Figure 9: Exemples of graded mesh with geometric refinement. At left: the mesh is refined on the west boundary only. At right: refinement at both west and east boundaries.

Here are the main options of the Edge Sizing details panel.

- Number of Divisions corresponds to the number N of vertices of the edge.
- Choose Behaviour: Hard to disable global sizing options.
- Bias Type enables to create a graded mesh with geometric refinement of the form $\Delta x_{i+1} = R\Delta x_i$, where the Bias Growth Rate R is the ratio between successive cells. Examples of graded meshes are given in Fig. 9.
- Bias Option: the option Smooth Transition needs the value of R, otherwise Bias Factor corresponds to the ratio between the maximal and minimal cells of the edge, ie: $\Delta x_{\max}/\Delta x_{\min} = R^{1/(N-1)}$.

4.2 Definition of the boundary conditions

Now that the meshing of the computational domain is completed, the last step before starting the CFD analysis with the Fluent solver is to label the boundary conditions as in Fig. 3. To perform this, click on Model in the Tree Outline, then select a boundary edge in the Graphics Window and insert a Named Selection[8]. In the Tree Outline, rename it as shown in Fig. 3. The label of the boundary edges will be recognized by Fluent and the corresponding boundary condition will be applied.

For an easy reviewing of the boundary conditions in Fluent by the user (think that an actual 3D mesh may have hundreds of boundary edges), it is worthwhile to group all edges with the same boundary conditions (*e.g.* the 3 edge walls) in a single named selection.

Question 3. *Once the mesh is created and the boundary conditions labeled, display the mesh in your* Word *report, give the mesh statistics and exit* Ansys Meshing .

[7] Right-click on Mesh in the Tree Outline and select Clear Generated Data. This operation only deletes the mesh, not the meshing methods you have defined.

[8] Right-click on the moose: Insert → Named Selection.

5 CFD analysis with Fluent

Open Fluent from WorkBench . The Fluent Launcher panel will open. Select

Dimension : 2D, Solver Options : Double Precision

and leave the default settings for the other options. Note that:

- Solver Options : Choosing Double Precision allows floating point operations with a 16-digit precision. It is not recommended to use Single Precision because less accurate computations (8-digit accuracy) may impact the convergence of the iterative solution, even if it reduces the memory usage and yield faster computations.
- Parallel (Local Machine) : Depending on the number of processors (cores) available on your PC, you can select the number of processes used by Fluent to perform the computations in parallel (see Fig. 10). This exercise can be performed in serial (Solver Processes=1). In an ideal case, a parallel computation (Solver Processes ≥ 2) allows to divide the time of a serial computation by the number of processors.

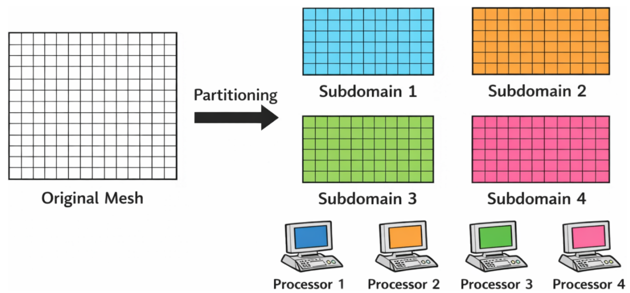


Figure 10: Example of mesh partitioning for parallel computation (MPI protocol). The computation in each of the 4 subdomains is performed in a separate processor.

5.1 Overview of Fluent

The graphics interface (GUI) of Fluent is shown in Fig. 11. The most relevant components are:

- Toolbars: As in DesignModeler and Ansys Meshing .
- Graphics Window: Graphical results are displayed here. The default mouse button functionality is[9]:
  - Left button translates.
  - Middle button zooms.

[9] Note that they are different from the other softwares in WorkBench . They can be changed from Display → Mouse buttons... in the menu bar.

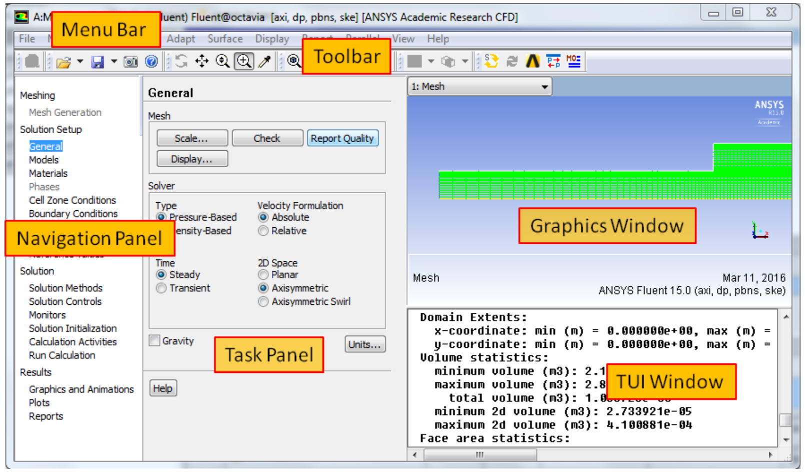


Figure 11: GUI of Fluent .

- Right button selects/probes.

- Navigation Panel: The main stages of the CFD analysis are displayed on the left of the GUI from top to bottom:
  - Solution Setup: define the CFD problem, such as the fluid models (laminar, turbulent, heat transfer, multiphase, . . .), type of materials (water, gas, solid, . . .), boundary conditions.
  - Solution: define the numerical inputs (such as discretization schemes, solution algorithms, initial condition, . . .) and solve the problem.
  - Results: once the solution is calculated, analyze (postprocess) the results.
- Task Panel: Selecting an item in the Navigation Panel opens the corresponding dialog box in this panel.
- Menu Bar: Additional items can be accessed from the menu bar.
- TUI[10] Window: display text results of the computation.[11]

5.2 Input/output files

There are 2 types of data files generated by Fluent .

- Case File *(*.cas)* : stores the mesh, the setup and the solution parameters of the simulation.

[10]TUI: Text User Interface.

[11]For advanced user: all the tasks defined from graphic panels have a corresponding TUI command, in order to be able to run Fluent in batch mode for automating the solver set up/solution/postprocess stages with scripts and journal files.

- Data File *(*.dat)* : stores the last computed solution.

You can manage these files from the menu bar File → Import or Export. You can also import a mesh generated from a third-party meshing software. Since you are using WorkBench , the mesh generated previously with Ansys Meshing is imported at the start of Fluent , while Case and Data files are automatically managed when you start or exit Fluent .

5.3 Solution setup

All the Fluent tasks for set-up of the CFD problem are done in the Solution Setup panel. Fluent has predefined default settings for the various stages of the set-up, but it is important to set them correctly for the problem under consideration. Thus, it is important to have knowledge of the physics involved in your CFD problem for selecting the relevant flow simplifications, models and equations to be solved.

Question 4. *We consider the flow of water ($\rho = 998.2\,kg/m^3$, $\mu = 0.001003\,kg/ms$) with inlet velocity $u_{inlet} = 1.5 \times 10^{-3}\,m/s$. Compute the Reynolds number in the upstream pipe (based on its diameter). Infer the flow regime and the relevant flow simplifications.*

Check the mesh and select the CFD solver: General task page

It is important to check the suitability of the imported mesh, as previous WorkBench operations may have created small errors that render the mesh unusable for Fluent computations.

Question 5. *Check the mesh by clicking on* Check *on the* Mesh *subpanel. Verify that the domain dimensions are the same as in Fig. 3 and report the mesh characteristics on your* Word *report.*

If the mesh check has failed, it may be caused by the fact that part of the domain falls below the upper half plane, see Sect. 3. A simple way to fix this problem is to translate the mesh[12] such that the symmetry axis is set at $y = 0$ (or slightly above).

As mentioned in the CFD lectures, there is no universal numerical method that works for all flow regimes. Choose the Solver Type according to:

- Incompressible flows are computed with the Pressure-Based Solver,
- Compressible flows at high Mach numbers are computed with the Density-Based Solver.

The other options are related to the problem simplification mentioned in Section 1: Time: Steady, 2D Space: Axisymmetric.

For large simulations, the computation time and the memory usage can be reduce by reordering the cell numbering of the mesh constructed by Ansys Meshing , in order to improve the solution of algebraic systems. To perform the domain reordering in Fluent , type the following command in the console window :

```
/mesh/reorder/band-width/reorder-domain
```

It will allow faster and more efficient computations, but will not affect the final results. Note that this reordering is done by default in parallel computations.

[12]Do it with Mesh → Translate. . . in the menu bar.

Set up the flow physics: Models task page

The Pressure-based solver is now configured with the relevant flow physics in the Models task page: turbulence models, flow with heat transfer (Energy, Radiation), combustion (Species), multiple phases (Multiphase), ...

It is of the upmost importance to select the model of the viscous stresses of the Navier-Stokes equations (Viscous item). Click on the Edit... button to review the various Viscous options:

- Inviscid : viscosity is assumed to be zero, which corresponds to the Euler equations, mainly used for external compressible flows.
- Laminar : corresponds to the classical Navier-Stokes equations with the Cauchy stress tensor. The viscosity model (constant, non-Newtonian or temperature dependent) is defined later in the Materials task page.
- Other options are used for turbulent computations, and correspond to choosing the modified viscous tensor resulting from RANS[13] modeling of the Navier-Stokes equations. We can mention the well-know k-epsilon model which is widely used in CFD: using this model will add 2 transport equations (for the turbulent kinetic energy k and dissipation ϵ) to the NS equations. The parameters for these transport equations can be set from these this panel.

The last versions of Fluent uses by default a RANS model with all other options (Multiphase, Energy, *etc...*) disabled. Check your models carefully according to the flow physics.

Set up the material properties: Materials task page

Materials refer to the various fluids or solids involved in your CFD problem. The materials available for your simulation are listed on the Materials task page. By default, Fluent selects the Fluid material as air with constant properties (density and viscosity). Click on Create/Edit... to add water-liquid to the list of available materials (use the Fluent Database, where a wide range of materials are predefined). You can edit the material properties if needed, for example if you wish to use a non-Newtonian fluid with shear-thinning viscosity.

Set up the cell zone conditions

As stated in the lectures, a zone corresponds to a portion of the mesh defined as fluid, solid or porous, where distinct materials can be defined[14]. These zones have been defined previously in DesignModeler : in this first lab exercise, there is only one fluid zone filled with water. Click on Edit... in the Cell Zone Conditions task page to set this zone as water-liquid (by default Fluent fills all fluid zone with air).

[13] *RANS : Reynolds Averaged Navier-Stokes Equations*, obtained through statistical modeling of turbulent flows.

[14] As an example, the computational domain for a conjugate heat transfer problem would be divided into fluid and solid zones.

Set up the boundary conditions (BCs)

This task page defines the BCs settings at the boundary edges (in 2D) of the computational domain. The Zone panel displays the name of the boundary edges as you previously labelled them in Ansys Meshing [15]. You can visualize theses edges by clicking on Display Mesh.... Fluent usually recognizes these labels and assigns them with a corresponding BC Type. Make sure the the inlet boundary is defined as velocity-inlet, the symmetry axis as axis, and the outlet as pressure-outlet (see Fig 1). Some BCs needs additional settings which are defined by clicking on Edit.... For this lab exercise, you only have to set the inlet velocity given by Question 1.

5.4 Set up the numerical parameters of the computation

Now that the physical model, solver and boundary conditions have been set, the next step is to define the numerical parameters (algorithm for pressure-velocity coupling, spatial discretization, convergence criteria) of the pressure-based solver: this is performed with the Solution panel.

By default, Fluent chooses “robusts” numerical methods and parameters, which yield converged solutions for a wide range of flows. Henceforth, the default discretization methods are usually low-order and dissipative (for stabilizing the solution) and should be modified to get accurate results. The following sections gives guidelines for choosing these numerical methods. Since these methods are improved and their range of applicability expanded at each yearly releases of Fluent , the recommendations given at the time of writing (updated winter 2016) may be obsolete in the following years.

Set up the discretization methods: Solution methods task page

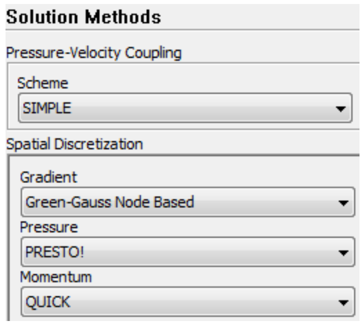


Figure 12: Recommended discretization methods for the expansion flow.

In this task page (shown in Fig 12) you first set up the iterative algorithm for the pressure-velocity coupling :

- SIMPLE[16] is the original algorithm for incompressible flow computations, still nowadays still in use for steady flows (single or multiphase).

[15] If some of the boundary edges are missing, it may be caused by a bug in the WorkBench . Ask advice to your instructor.

[16] Acronym for *Semi-Implicit Method for Pressure-Linked Equations.*

- Prefer PISO for time-dependent computations.
- Starting from ANSYS V19, the Coupled algorithm is proposed as default. This algorithm may reduce the computational time of SIMPLE or PISO, but has not been fully tested for all the physical models of Fluent .

Note that the use of a particular pressure-velocity algorithm does not change the final solution if convergence is reached. However, the convergence properties of the iterative solution (numerical instabilities, convergence rate and simulation time) are strongly depend to the chosen algorithm.

Next, define the various finite-volume (FV) discretization schemes for the NS equations[17] (Gradient: cell-face interpolation of velocity gradient used in the discretization of the viscous term and high-order discretization of the convection term, Momentum: discretization of convection term, Pressure: interpolation of cell-face pressure). The various pros and cons of each scheme has been discussed in the lecture notes. As a reminder:

- Avoid Green-Gauss Cell-Based interpolation which is the least accurate.
- Use PRESTO! for pressure interpolation in incompressible flows, or the solution may have convergence problems.
- For convection, use First Order Upwind solely at the initial stages of the computation or if the solution exhibits convergence problems. Otherwise, second-order schemes such as QUICK (more accurate and less diffusive) are preferred.

Set up the parameters of the iterative algorithm: Solution Controls task page

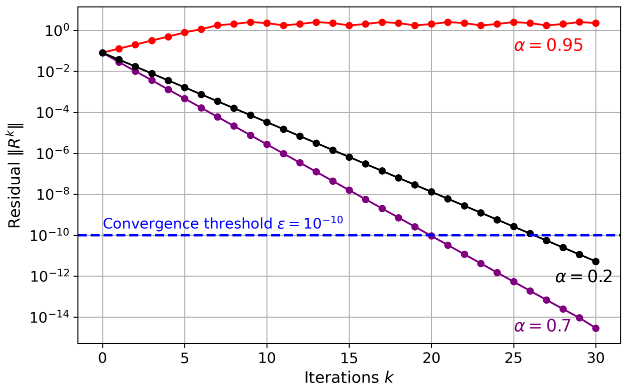


Figure 13: Effects of α on the convergence of an iterative algorithm for solving (1).

The basics of the iterative solution of agebraic equations in Fluent is explained for the linear system :

$$\mathcal{A}\Phi = F, \tag{1}$$

[17] When applicable, discretization schemes of the additional transport equations need to be defined too. Examples will be given for the turbulent computation of Section 6.

where Φ is the vector containing velocity or pressure mesh values, the matrix $\mathcal{A}$ and vector F represent the FV discretization of flow equations.

Starting from the Initial Solution Φ^0, an iterative algorithm seeks the solution to (1) as the series $\{\Phi^k,\, k = 1, \ldots, k_{\max}\}$ defined as:

$$\Phi^k = \Phi^{k-1} + \alpha\,\delta\Phi, \tag{2}$$

where k is the iteration number, $k_{\max}$ in the maximal number of iterations, and $\alpha \in [0, 1]$ is the Under-Relaxation Factor. Fluent considers that convergence is reached when the residual R^k of Eq. (1), defined as $R^k = F - \mathcal{A}\Phi^k$, verifies the Criterion:

$$\|R^k\| < \epsilon. \tag{3}$$

For the iterative solution of each flow equations (Momentum, Pressure, ...), you have to input the value of following parameters in Fluent :

- The initial condition[18] Φ^0,
- The convergence threshold ϵ,
- The under-relaxation factor $\alpha \in [0, 1]$,
- The maximal number of iterations $k_{\max}$.

The value of the under-relaxation factor α has impact on the rate of convergence of the algorithm, as illustrated in Fig. 13 : a large value of α gives faster convergence (compare $\alpha = 0.7$ and $\alpha = 0.2$), but a value too close to 1 yields non-convergence ($\alpha = 0.95$).

The Under-Relaxation Factors are set in the Solution Controls task page for each flow equation. The default values given by Fluent are *robusts*, *i.e.* they give a solution for a wide range of applications. There is no need to modify them unless your computation undergoes convergence problems : in this case, lower the under-relaxation factor of the equation that is not converging.

The set-up of Φ^0, ϵ and $k_{\max}$ is described next.

Monitoring the convergence of the iterative algorithm: Solution controls task page

During the simulation, Fluent allows you to monitor the evolution of the residual R^k in a graphic window as in Fig. 14. The monitoring parameters of each flow equations are set in the Monitors task page. By default, Fluent defines the Absolute Criteria of (3) as $\epsilon = 10^{-3}$ for all equations[19]. Often in practical applications, this condition is not sufficiently restrictive to give an accurate solution. To change this criterion, click on Edit... in the Residuals panel (see Fig. 14).

5.5 Perform the flow computation

The computations are performed in the Solution panel. You need first to define the initial condition Φ^0 of the iterative algorithm. This is performed in the Solution Initialization task page. Fluent proposes several Initialization Methods for defining suitable initial conditions, notably from the values defined at the boundary conditions. For this lab exercise, you will start the algorithm from rest ($\boldsymbol{v} = 0$, $p = 0$). To do this, choose Standard Initialization in the task page,

[18] For unsteady problems, the initial condition corresponds to the solution at initial time $t = 0$.
[19] Except the Energy equation for heat transfer applications where it is set as $\epsilon = 10^{-6}$.

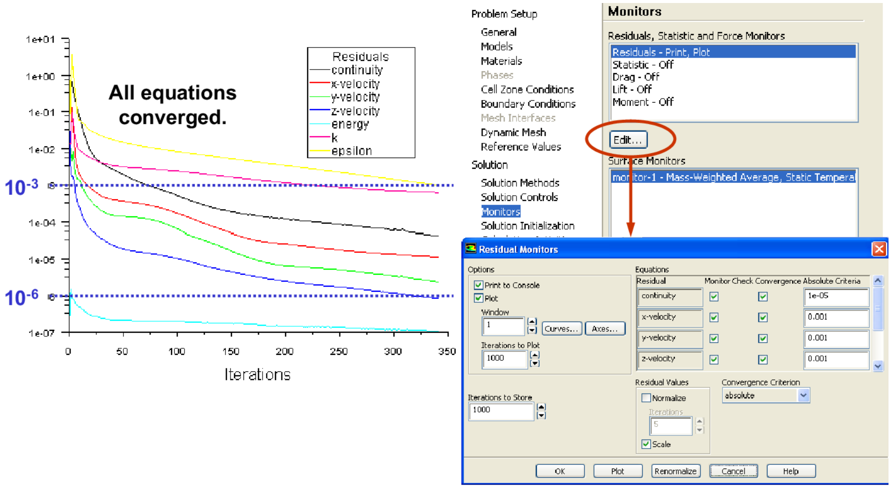


Figure 14: Example of residual monitoring plot for a non-isothermal turbulent flow computation.

Compute From: (all-zones), and press Initialize. Optional : the initial solution is ready to be visualized by the postporcessing tools of Section 5.6.

To start the iterative algorithm, go to the Run Calculation task panel, enter a non-zero value for the (maximal) Number of Iterations $k_{\max}$ and press (Calculate). As the computation is performed, Fluent plots in a graphic window the residuals at each iteration[20]. The computation will stop when the convergence criteria (3) is reached or $k > k_{\max}$, and the text output in the TUI will be something like :

```
 iter  continuity  x-velocity  y-velocity     time/iter
   470  8.2896e-06  2.4845e-06  9.6828e-08  0:00:07  530
!  470 solution is converged
```

If the last line "`solution is converged`" was missing, then convergence is not reached because the value of $k_{\max}$ is too low. In that case, input a larger value for $k_{\max}$ and press (Calculate) to resume the computation. If you want to restart the algorithm from the start ($k = 0$), redo the Solution Initialization before clicking on (Calculate).

Question 6. *Perform a computation of the flow with the convergence threshold $\epsilon = 10^{-5}$ for all flow equations. Verify that the solution is converged. Display the residual plot and the text output in your* Word *report*[a].

[a]For exporting a Fluent plot, right-click at the top of the graphics window and select Copy to Clipboard.

For advanced users: to perform periodically a back-up copy of the solution during the iterative algorithm, go to the Calculation Activities panel and enter the frequency back-up in the Autosave Every (Iterations) group box.

[20]The frequency of the writing and plotting can be changed with the Reporting Interval option.

5.6 Postprocessing: Results panel

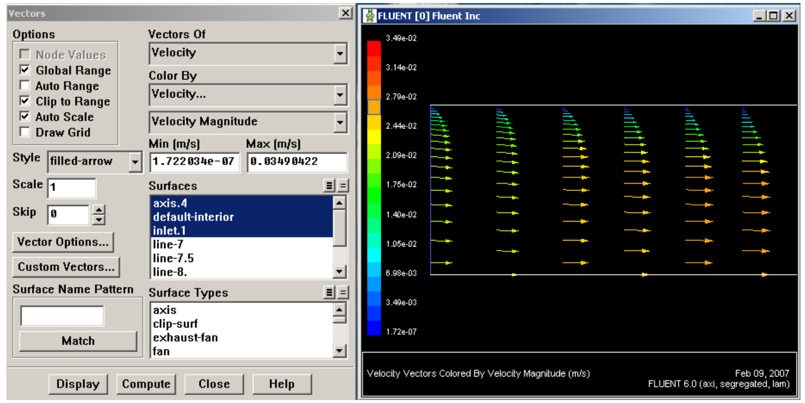


Figure 15: At left: Vectors dialog box. At right: 2D vizualisation of the velocity vectors upstream of the contraction.

Once the converged solution is obtained, Fluent proposes a variety of postprocessing tools to analyze the solution in order to extract relevant flow information. This is performed under the Results panel. There are 3 main types of analysis:

- 2D (or 3D) solution visualization (Graphics and Animation task panel): for plotting the velocity vectors, contour levels of a flow variable (velocity, pressure) inside the computational domain.
- 1D visualization (Plots task panel): for profile plotting of the solution on selected sections (or lines) of the computional domain.
- Numerical reporting of mean flow quantities (Reports task panel): such as mass flow through selected sections, forces at boundaries, . . .

Velocity vector visualization: Graphics and Animation task panel

Choose Vectors in the Graphics group box and click on (Set Up...). The Vectors dialog box of Fig. 15 (left) opens. From the Surfaces drop-down list choose interior to display the velocity vectors in the whole computational domain[21], and press (Display). You can use the mouse to zoom in/out in the graphics window.

Question 7. *Zoom in the computational domain to observe the recirculation zone of the flow (see Fig. 1). Use the* Scale *options of the* Vectors *dialog box if the vectors are too small. Display this plot in your* Word *report.*

[21]Alternatively, you can visualize the flow on selected surfaces and boundaries from the Surfaces drop-down list.

Isocontour visualization: Graphics and Animation task panel

Choose Contours in the Graphics group box and click on $\boxed{\text{Set Up...}}$ to open the Contours dialog box. Select a flow variable in the Contours of drop-down list and press $\boxed{\text{Display}}$.

Most flow variables are defined in Fluent with intuitive names. For example, for asymmetric flow:

$$u = \text{Axial Velocity}, \qquad v = \text{Radial Velocity}, \qquad ||\boldsymbol{v}|| = \text{Velocity Magnitude}.$$

The physical pressure (used in thermodynamics) is the Absolute Pressure $p_{\text{abso}}(\boldsymbol{x}, t)$, which is

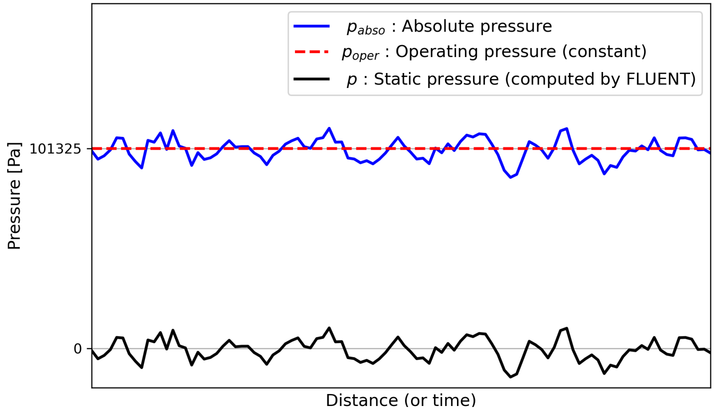


Figure 16: Illustration of the pressure decomposition (4) used in Fluent .

decomposed in Fluent as :

$$p_{\text{abso}}(\boldsymbol{x}, t) = p(\boldsymbol{x}, t) + p_{\text{oper}}, \tag{4}$$

where :

- p_{oper} : the Operating Pressure is defined as a constant in the $\boxed{\text{Operating Conditions...}}$ panel, and is set by default as the atmospheric pressure $p_{\text{abso}} = 101325\,\text{Pa}$. Modifying p_{oper} is only significant for other regimes than incompressible flows.
- $p(\boldsymbol{x}, t)$: the Static Pressure (or gauge pressure) is the field actually computed by Fluent .

This decomposition makes sense in an incompressible flow where the static pressure shows small deviations from the atmospheric pressure (see Fig. 16). Other useful quantities related to pressure are :

- $p_{\text{dyn}} = \frac{1}{2}\rho\,||\boldsymbol{v}||^2$: the Dynamic Pressure,
- $p_{\text{total}} = p + \frac{1}{2}\rho\,||\boldsymbol{v}||^2$: the Total Pressure, useful for calculating head losses.

Note that in incompressible flows :

- Density ρ is constant so the pressure is not used for its calculation[22],

[22] In compressible flows : $\rho = p_{\text{abso}}/rT$; in incompressible ideal gas flow : $\rho = p_{\text{oper}}/rT$.

- Only pressure differences are relevant for calculating head losses and hydrodynamic forces, which can be performed in Fluent with the Static Pressure $p(\boldsymbol{x}, t)$.

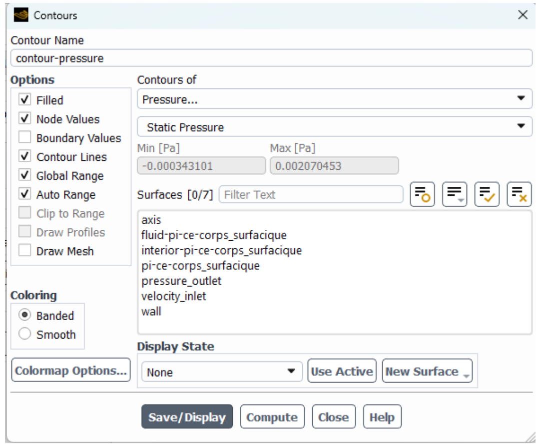


Figure 17: Recommended settings for pressure contour visualisation (choose less than 50 contour levels). It is possible to modify the level range by deselecting the Auto Range option and choosing manually the Min and Max values.

Question 8. *Display the isobaric (pressure) lines of the flow near the contraction. The recommended settings for pressure visualisation are shown in Fig 17 (You can modfiy the number of contours in the* $\boxed{\text{Colormap Options ...}}$*). Export the figure to your* Word *report. Compare the isobaric contours downstream of the singularity with the theoretical pressure of fully developped Hagen-Poiseuille flow.*

Velocity profiles on sections of the computational domain

This part concerns the plotting of velocity vectors on selected surfaces (lines in 2D) of the computational domain, for example on the $x = 7\,\text{m}$ section[23] near the outlet boundary, in order to verify that the flow is full developed there. To do this, select $\boxed{\text{Surface}}$ → Line/Rake... from the menu bar to create the line $x_0 = 7.0$, $y_0 = 0$, $x_1 = 7.0$, $y_1 = 0.2$. Give the line a meaningful label, such as line-7.0 in New Surface Name. The surface line-7.0 will now be available in the Surfaces drop-down list of any Graphics and Vizualization dialog box.

Question 9. *Plot the velocity profiles on Sections* $x = 6.0, 6.5, 7., 7.5$ *and* pressure_outlet.* *(Give them a meaningful name!). Check that the Poiseuille velocity profile is obtained on these sections. Export the graphics to your* Word *report.*

[23] Remember that the origin $x = 0$ correspond to the section of the abrupt expansion.

Plot 1D profiles

The plotting of 1D profiles of predefined flow variables (axial or radial velocity, static or total pressure, ...) is performed on the Plots task panel. Choose XY Plot in the Plots group box and click on Set Up... to open the Solution XY Plot dialog box. For plotting along the radial direction y, choose the direction vector $\mathsf{X} = 0,\ \mathsf{Y} = 1,\ \mathsf{Z} = 0$ in the Plot Direction group box.

> **Question 10.** *Plot the velocity and (static) pressure along the sections of Question 9. Export the graphics to your* Word *report. Do these profiles correspond to a fully developed Poiseuille flow ?*

Define custom field functions

It is possible to define new flow variables (named "custom field functions") from Fluent predefined variables, by using the "pocket calculator" provided by Define → Custom Field Functions... from the menu bar. This new variable can be visualized as previously.

> **Question 11.** *Define the new function* tau_xy *that calculates the shear stress in cylindrical coordinates :*
> $$\tau_{xy} = \mu\Big(\frac{\partial v}{\partial x} + \frac{\partial u}{\partial y}\Big).$$
> *Plot the shear stress profile on the vertical sections of Question 9 and along the upper wall. Export the graphics to your* Word *report.*

Calculate mean values

For example, you can calculate the mean flow velocity in Section $x = 8\,\mathrm{m}$ of the pipe. In the Reports task panel, open the Surface Integrals dialog box and choose Area Weighted Average in the Report Type drop-down list.

> **Question 12.** *Verify that the mean flow velocity $\overline{V}$ is constant on the sections of Question 9. Report these values in your* Word *report.*

The Flux Reports dialog box allow you to calculate fluxes at selected surfaces of the domain.

> **Question 13.** *Use the Mass Flow Rate report to verify the mass flow conservation between the inlet and outlet boundaries.*

5.7 Application : calculation of standard pressure drop

In a duct of diameter D, the pressure (or head) loss due to viscous friction between sections x_1 and x_2 reads :

$$H_1 - H_2 = \lambda \frac{\overline{V}^2}{2g}\frac{x_2 - x_1}{D}, \tag{5}$$

where λ is the (dimensionless) friction factor, $\overline{V}$ is the mean flow velocity, g is the gravity, and $H_i = \overline{p}_{\text{total}}(x_i)/\rho g$ is the average head at Section x_i. This formula is valid when the streamlines

are parallel to each others, *i.e.* when the flow is fully developed in the pipe.

> **Question 14.** *According to the previous questions, the flow is fully developed between the zones $6.5 \lesssim x \leq 8$ of the pipe. Use* Fluent *to report the total pressure $\overline{p}_{tot}$ in various sections of the fully developed region, then use (5) to calculate the corresponding value of λ. Compare your results with the empirical relations obtained in a straight pipe:*
>
> - *Poiseuille formula for laminar flows, $Re_D \lesssim 2000$:*
> $$\lambda = \frac{64}{Re_D},$$
> - *Blasius formula for turbulent flows, $4\times10^3 \lesssim Re_D \lesssim 10^5$:*
> $$\lambda = 0.3164 Re_D^{-1/4}.$$

6 Numerical computations in the turbulent regime

Before starting the turbulent computations, it is recommended to close FLUENT and make a copy of your laminar flow project in WorkBench in case of later use.

6.1 Performing the computation

We now consider the flow of water with inlet velocity $u_{\text{inlet}} = 1.5 \times 10^{-1}\,\mathrm{m/s}$.

> **Question 15.** *Calculate the Reynolds number. Deduce the flow regime and the flow equations to solve.*

If a turbulence model is needed, use k-epsilon Standard. In addition to the averaged momentum and continuity equations, transport equations for k and ϵ are now solved. You need to define the settings for these additional equations, in particular:

- For inlet BCs, choose Intensity and Hydraulic Diameter in the drop-down list Turbulence Specification Method, and set a turbulent intensity of 5% and an hydraulic diameter of 0.2 m.
- Use 2nd-order methods for the convective terms of these equations.
- Define the tolerance monitoring parameters for these equations.

> **Question 16.** *Starting from rest, compute the solution with residual threshold $\epsilon = 10^{-6}$ for all equations. Export the residual history to your* Word *report.*

When performing a high-Reynolds number with impulsive start from the rest, the iterative solution may undergo convergence problems: the residuals do not decrease monotonously, stall or increase until solution blow-up. If this happen, you need to modify the default solver settings for obtaining convergence. The "optimal" parameters for convergence are usually problems dependent, and fine-tuning of these parameters requires expertise and trial-and-errors. General guidelines for convergence troubleshooting are given in the lecture notes. Here is a quick summary:

– Initialize the solution with a previously calculated solution at lower Reynolds number, then gradually increases the inlet velocity.
– Use first-order methods in the initial stages of the iterative algorithm, then use more accurate methods.
– Decreases the values of the under-relaxation factors of the equation(s) that cause(s) difficulties (usually, it is the continuity equation).
– Improve the mesh quality and use finer cells in zones of high flow gradients.

However, for this lab exercise you shouldn't experience convergence problems: After an initial stage where the residuals seem to be stalled, the solution should converge after a few thousand iterations. Note however that this behaviour may depends on the discretization parameters you chose or the version of Fluent you are using.

6.2 Analyze the results

General flow features

Question 17. *Verify that the flow is fully developed at the outlet. Display in the 2D domain the velocity profiles near inlet (several sections in the upstream pipe) and outlet (sections of Question 9). Compare with the laminar velocity profiles obtained previously. Report these results in your* Word *report.*

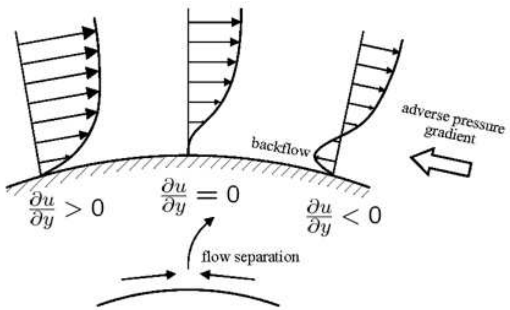


Figure 18: Velocity profiles at a wall boundary layer near a separation point.

Calculation of singular head losses

At the vicinity of a geometrical singularity such as the sudden expansion at $x_1 = 0\,m$ (*cf.* Fig. 1), the flow is no more unidirectional and the head loss equation (5) is not valid anymore. Experiments show that singular head losses are proportional to the mean kinetic energy:

$$\Delta H = \frac{\xi}{2g}\overline{V}^2,$$

where ξ is a dimensionless coefficient that depends on the type of singular geometry. For singular expansion flows, the value of the head-loss coefficient ξ can be estimated analytically at high flow rates with the following assumptions:

- Losses due to wall friction are negligible,
- The streamlines are parallel through Section S_1 ($x_1 = 1.0\,m$) and Section S_2 (*cf.* Fig. 1), where the flow returns to fully developed regime. This implies that pressure distribution is uniform at these sections.

With these hypotheses, conservation of mass and momentum between S_1 and S_2 yields:

$$H_1 - H_2 = \frac{\xi}{2g}\overline{V}^2(x_1), \qquad \text{with} \quad \xi = \left(1 - \frac{S_1}{S_2}\right)^2. \tag{6}$$

This formula can be found in any fluid dynamics textbook.

Question 18. *From the head losses given by your numerical solution, calculate the head-loss coefficient ξ and compare with the analytical value. The difficulty is to estimate the abscissa x_2 where the streamlines do reattach. To estimate x_2, you may use the property that the shear stress is zero at a separation point (where flow separates or reattaches, see Fig. 18): the value of x_2 is then obtain by plotting the computed shear stress along the upper wall (cf. Question 11).*

Lab Exercice on Computational Fluid Dynamics

Unsteady flow past circular cylinder with ANSYS FLUENT

Instructor : Olivier Botella olivier.botella@univ-lorraine.fr

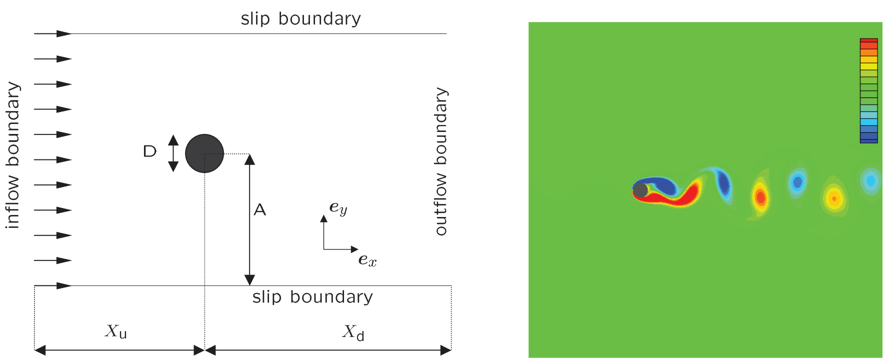


FIGURE 1 – *At left* : Computational domain and boundary conditions. The domain dimensions are : $D = 0.1$ m, $X_u = 0.8$ m, $X_d = 1.5$ m, $A = 0.8$ m. *At left* : instantaneous vorticity field $\omega = \partial v/\partial x - \partial u/\partial y$ at $Re = 100$.

1 Problem statement

This lab exercise concerns the CFD analysis of the laminar flow past a circular cylinder, see Fig. 1. The working fluid is air, and the Reynolds number of the flow is :

$$\mathrm{Re} = \frac{\rho U_\infty D}{\mu},$$

with the velocity at the inlet U_∞ as reference velocity, and the cylinder diameter D as reference length.

At low Reynolds number the flow is steady, and above the critical value $\mathrm{Re}_c \simeq 47$ the flow undergoes a Hopf bifurcation that causes unsteady oscillations of the wake : the well-known *von Kàrmàn vortex street* (*cf.* Fig. 1). In the periodic regime ($47 \lesssim \mathrm{Re} \lesssim 194$), the frequency f of the vortex shedding is usually given as the dimensionless *Strouhal* number St :

$$\mathrm{St} = \frac{D}{U_\infty} f.$$

The Strouhal number is obtained by calculating the drag $F_x(t)$ and lift forces $F_y(t)$ exerted by the flow at the cylinder surface, and by performing a *FFT*[1] of the time-signal when the

1. FFT : Fast Fourier Transform.

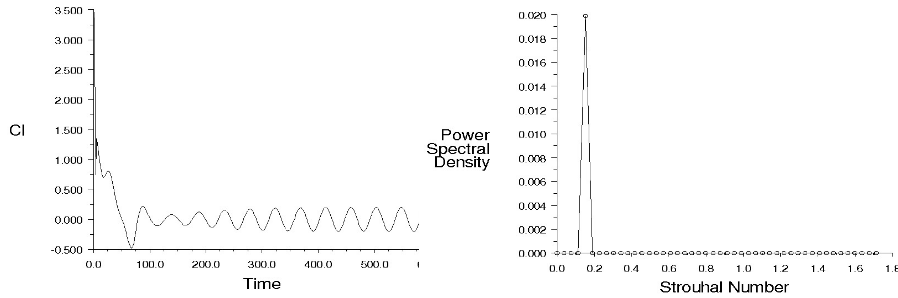


FIGURE 2 – *At left* : Time history of the lift coefficient C_L (the time is in seconds). *At right* : FFT of the lift signal in the time interval $t \in [400, 600]$.

flow has asymptotically reached its periodic regime, *cf.* Fig. 2. The hydrodynamic forces at the cylinder are usually given as the dimensionless *drag and lift coefficients* :

$$C_D(t) = \frac{F_x(t)}{\frac{1}{2}\rho U_\infty^2 S}, \qquad C_L(t) = \frac{F_y(t)}{\frac{1}{2}\rho U_\infty^2 S},$$

where S (m^2 in SI units) is the projected area of the cylinder [2].

The purpose of this lab exercise is to compute the flow at Re = 100 with ANSYS Fluent , to study the vortex shedding and compare the values C_D, C_L et St given by your computations with the reference values of Table 1.

St	$\overline{C}_D$	$C_{L_{\max}}$
$0.16 - 0.17$	$1.21 - 1.41$	$0.31 - 0.40$

TABLE 1 – Experimental and computational reference data of the flow at Re = 100. $\overline{C}_D$ denotes the mean value of the drag coefficient and $C_{L_{\max}}$ is the maximal value of the drag coefficient, calculated when the flow reached its periodic state.

2 Geometry creation with DesignModeler

The computational domain to create is shown in Fig. 1. The origin of the domain is the cylinder center, and for obtaining accurate numerical results it is important that the domain be symmetrical with respect to the horizontal axis. To create this domain, we consider that the fluid domain consists in a rectangle from which a circle is subtracted (with a boolean operation). For this reason, it is worthwhile to draw the rectangle and the circle in 2 different sketches.

2. The projected area S is the frontal surface of the body projected on the spanwise direction. For 2D flow past circular cylinder of diameter D, we get $S = D$.

0) Go to the Settings toolbox to apply the adequate parameters for sketching : Major Grid spacing : 0,1 m, Minor-steps per Major : 10. For ease of sketching, show the grid and toggle the Snap option so the drawings will be aligned to the grid lines.

1) In a first sketch, create a rectangle with bottom-left corner of coordinates (-0.8 m, -0.8 m). It is then easy to center the rectangle with respect to the origin and give its correct dimensions. In a second sketch, create the circle [3].

2) Create the surface sketch from these 2 sketches with the Add Frozen option. Rename the first surface body as "fluid" and the second as "cylinder".

3) Now, you have to subtract the cylinder body to the fluid body by a boolean subtraction (Create → Boolean from the menu bar). Use the following details : Operation : Subtract, Target body : Fluid, Tool body : Cylinder, Preserve Tool Body : No.

After application of the boolean operation, you should have created 1 part, with 1 "fluid" body. Set its Fluid/solid detail to Fluid, save the project and exit DesignModeler .

3 Mesh creation with Ansys Meshing

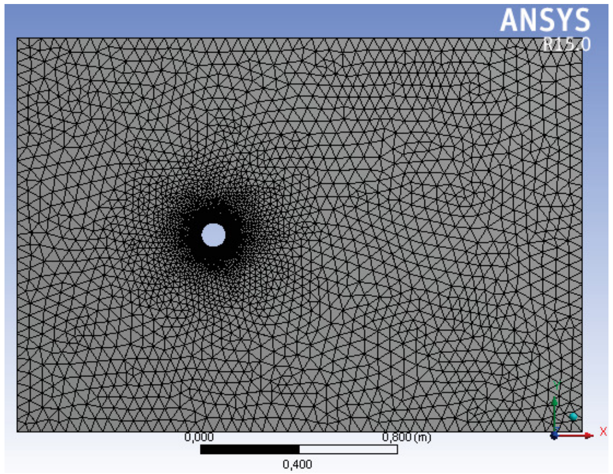


FIGURE 3 – Triangle mesh generated with the curvature size function and edge sizing of the cylinder surface. This mesh has 9715 cells.

The purpose of this part is to generate the unstructured triangle mesh shown in Fig. 3, which is refined around the cylinder surface in order to resolve the fine vortical structures in the boundary layers and the wake. To do so, Ansys Meshing use the following strategies :

– First, you define a Global Meshing Method in order to define the type of elements used (quad, tri, ...) and the average size of the elements. In addition, DesignModeler use a "size function" to automatically control the mesh density in the vicinity of highly curved boundaries by using various strategies (proximity, curvature, ..., see the CFD lectures). These global mesh parameters are set in the Details View of the mesh. By default, the size function is enabled (Use Advanced Size Function : On :Curvature).

– Secondly, you may ameliorate the mesh in specific areas by inserting Local Meshing Controls such as edge or face sizing, inflation, *etc.*..(see CFD lectures). This step is optional if you only need a crude mesh, but is required to obtained quality meshes.

3. If needed, use the Coincident constraint to put its center to the origin of the domain.

Following this strategy, the mesh is generated by the following steps :

1) Global Meshing controls : In the Tree Outline, select the mesh and insert a meshing method [4] for the fluid domain. In its Details View, choose Method : Triangles. Now, define the global meshing controls : in the Sizing Details of the mesh, choose Max Face Size : 0,05 m and Growth Rate : 1,10 for the size function.

2) Generate the mesh. You may observe that the size function has detected the cylinder surface as a highly curved boundary and has refined the mesh in its vicinity [5]. However, the mesh is not as fine near the boundary as the one in Fig. 3.

3) Add Local Meshing controls : insert an Edge Sizing method in order to control the cell distribution on the cylinder surface with Number of Divisions : 200. The mesh generated will be similar to the one in Fig. 3.

> **Question 1.** *Display a figure of the various meshes you created in your* Word *report. The final mesh, to be used for the CFD analysis, should have the boundary conditions of Fig. 1 labeled with* Named Selections *before exiting* Ansys Meshing .

4 Flow computation with Fluent

The unstructured mesh generated with DesignModeler is now used to compute the flow Re = 100 (*reminder* : the working fluid is **air**).

4.1 Step 1 : Solution setup

In this preliminary stage, you will define the flow physics (flow equations, boundary conditions, ...), set up the solver and the discretization methods. Follow the guidelines given in the first lab exercise. Only new steps will be detailed below.

Guidelines for solution setup

- Since the Re = 100 flow is time-periodic, use the Unsteady solver. The PISO algorithm for pressure-velocity coupling is recommended.
- For setting the Inflow BC, reply to the following question :

> **Question 2.** *Calculate the inflow velocity* U_∞ *corresponding the the* $Re = 100$ *regime.*

4. Alternatively, toggle the volume selection icon, right-click Insert → Method on the fluid domain.
5. You may also generate a mesh with Use Advanced Size Function : Off and observe the differences.

- Choose the most accurate and robust discretization methods, according to your experience with the first lab exercise.
- Select 2nd-order accurate time-stepping [6] otherwise your computation may blow up.

Set up the time history of the *drag* C_D and *lift* C_L

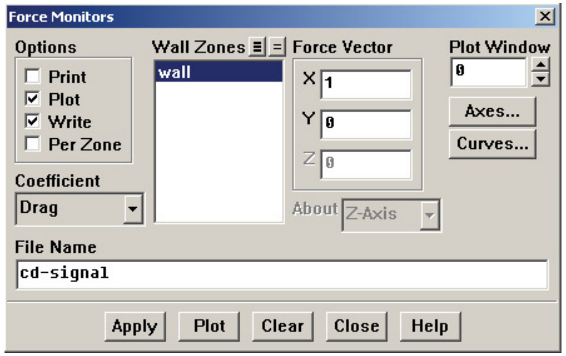


FIGURE 4 – Configuring the Drag Monitor dialog box. The drag is the horizontal component (Force Vector : $\mathbf{e}_x$) of the hydrodynamic force acting on the cylinder surface (Wall Zones). The Plot option will plot the signal in a graphical window during the computation, while the Write option will write the data in the text file cd-signal.

During the unsteady computation, Fluent is able to calculate on-the-fly the forces coefficients $C_\mathrm{L}(t)$ et $C_\mathrm{D}(t)$ in order to postprocess these results as in Fig. 2. Prepare these calculations with the following steps :

- Define Reference Values of the flow (Solution Setup panel). These references values will be used by Fluent in the postprocessing stage to calculate dimensionless flow quantities such as force coefficients.

> **Question 3.** *Define in* Fluent *the flow reference values given in Section 1. The projected area S is denoted* Area *in* Fluent .

- Set up the drag calculation with the Monitors task panel. Click on Create → Drag... and follow the instructions of the dialog box of Fig. 4. Do the same for the lift coefficient.

Setup the initial condition

In order to break the flow symmetry and trigger the Hopf instability, you will use the initial condition :

$$\boldsymbol{v}(\boldsymbol{x}, t=0) = \boldsymbol{v}_0 + \boldsymbol{v}_\mathrm{pert}(\boldsymbol{x}), \tag{1}$$

6. In the Solution Method Task Page, choose Second Order Implicit in the Transient Formulation drop-down list.

where $\boldsymbol{v}_0 = \mathbf{0}$ (fluid is initially at rest) and the perturbation $\boldsymbol{v}_\mathrm{pert} = u_\mathrm{pert}(y)\boldsymbol{e}_x$ such as :

$$u_\mathrm{pert}(y) = U_\infty \frac{y + |y|}{2y}. \tag{2}$$

In this fashion, the initial condition corresponds to the sheared velocity :

$$u(y, t=0) = \begin{cases} U_\infty & \text{if } y > 0 \text{ (upper part of the domain),} \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

This initial condition is defined in Fluent with the following steps :

- Define analytically $u_\mathrm{pert}(y)$ with the Custom Field Functions calculator [7].
- The initial condition is now defined with the Solution Initialization task panel. Define $\boldsymbol{v}_0 = \mathbf{0}$ in the fluid domain as in the first lab exercise. Then, add the perturbation $u_\mathrm{pert}(y)$ by "patching" it [8] to $\boldsymbol{v}_0$.

> **Question 4.** *Define this initial condition with* Fluent . *Plot the resulting velocity field in the whole computational domain and on a vertical section. Display these plots in your* Word *report.*

4.2 Step 2 : Flow computation

The task page for the unsteady solver is shown in Fig. 5. You need to define the size of the time-step Δt (in seconds) and the number of time-steps. These values should be chosen from numerical considerations (such as the CFL condition be respected for stable computations), and also from the timescales of the flow at $\mathrm{Re} = 100$, as done below.

Determine the time-step size

> **Question 5.** *Use the value of the Strouhal number (Table 1) to calculate the period T_{shed} (in seconds) of the vortex shedding. Deduce the value of Δt that allows about 20 time-steps during one vortex-shedding period.*

Determine the number of time-steps

> **Question 6.** *From the value of Δt and the relation $t_{max} = n_{max}\Delta t$ (in seconds), calculate the number of time-steps n_{max} for computing 600 seconds of the flow.*

7. Note that the y coordinate is given in the Field Functions drop-down list Mesh...
8. Click on Patch... to open the patch dialog box. Choose Variable : X-Velocity and Zones to Patch : fluid, then click on Patch.

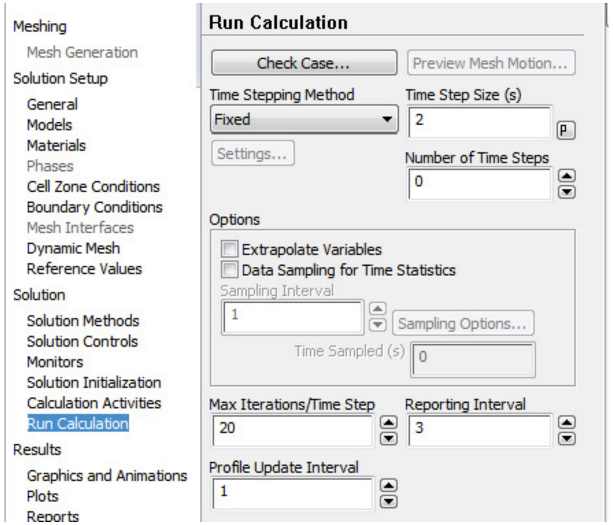


FIGURE 5 – Run calculation task page for the unsteady solver. The Fixed option uses a non-varying value of Δt (Time Step Size), necessary to perform a FFT of the time-signals. Max Iterations/Time Step defines the number of inner iterations of the velocity-pressure algorithm algorithm. Reporting Interval is the monitoring frequency of the algorithm.

Compute the flow

Question 7. *Click on* Calculate *to compute the flow. Check that the instability is triggered by monitoring the time evolution of the drag and lift coefficients that should be similar to Fig. 2. Export these plots to your* Word *report.*

Most Fluent solvers for incompressible flows[9] are fully implicit, yielding thus stable computations even at a CFL number greater than 1. Nonetheless, unstable computations may occur due to poor mesh quality, inadequate boundary conditions or stiff initial conditions. If solution blow-up happens in this lab exercise, review first your solution setup. If the computation is still unstable, restart the computation with a lower value of Δt for the first few time-steps, then increase Δt to its original value for the remaining of the computation.

It is advised to perform an autosave of the solution during time-integration, see first lab exercise.

4.3 Step 3 : Postprocessing

Now that the computation has been completed, this part is devoted to study the flow physics.

9. With the notable exception of Non Iterative Time Advancement and Projection Method.

Solution visualization

Question 8. *Export to your* Word *report the drag and lift history plots. Calculate $\overline{C}_D$ and $C_{L_{max}}$ when the flow reached the periodic regime, and compare your results with those of Table 1.*

Question 9. *Display the iso-vorticity lines (see Fig. 1) at a selected flow-time. In order to clearly visualize the cylinder wake, carefully choose an odd number of levels equally distributed around the zero isolevel (*Min $= -0.5$ *and* Max $= -0.5$*, for example). Select the option* Clip to Range *for better visualization.*

Unfortunately, only the norm of the vorticity, named Vorticity Magnitude[10], is defined in Fluent . You need to define the vorticity $\omega = \partial v/\partial x - \partial u/\partial y$ as a Custom Field Function in order to plot it.

Calculation of the Strouhal number

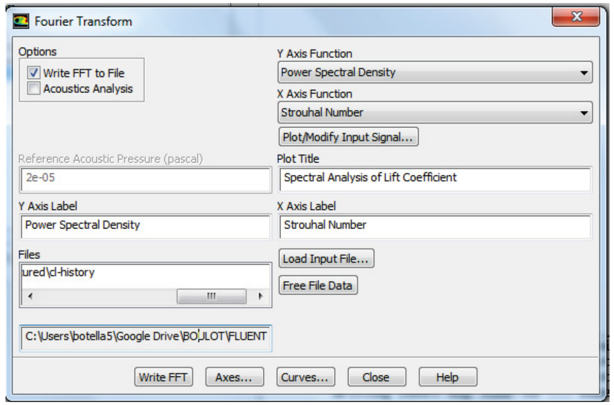


FIGURE 6 – Fourier Transform dialog box for FFT analysis of the lift coefficient.

The frequency of the vortex shedding can be estimated from the time history of the lift signal, *cf.* Question 8. However, it is customary to use signal processing tools such as the Fourier transform to precisely determine the flow regime (periodic, quasiperiodic, chaotic, ...). From the Plots task page, click on FFT to open the dialog box of Fig. 6, and follow these steps :

- Load the lift history file cl-signal.
- Use Plot/Modify Input Signal... to choose a time interval[11] where the periodic regime is reached, for example $t \in [400\,\mathrm{s}, 600\,\mathrm{s}]$.

10. In the Contours dialog box of the Graphics and Animation panel, Vorticity Magnitude is available in the Contours of Velocity... drop-down list.
11. We recall that the accuracy of frequency calculation with FFT is inversely proportional to the length of

- Click on [Plot FFT]

Question 10. *Export the plot of the FFT of the lift signal in your* Word *report. Calculate the frequency of the vortex shedding and give the incertitude of your calculation. Compare your results with the reference (Table 1).*

Graphical animation of the vortex shedding

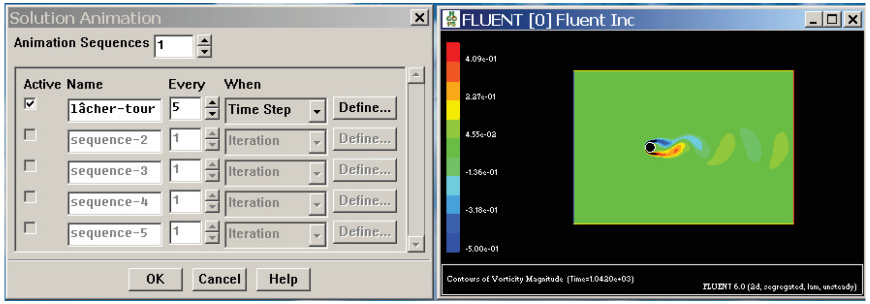


FIGURE 7 – *At left* : Solution Animation dialog box. *At right* : Iso-vorticity contours to be animated.

This part is devoted to the animation of the computed solution for visualization of the vortex shedding. This is done by performing a simulation on a few periods (for example $5\,T_{\text{shedd}}$ seconds) and periodically saving snapshots of the iso-vorticity lines on the disk (for example every 5 time-steps). The few dozen of snapshots can then be animated in a video-clip.

In the Calculation Activities task panel, open the Solution Animation dialog box of Fig. 7, and follow these steps :

- Activate an animation and label it. With the option [Define...] enter the iso-vorticity lines of Question 9.
- Resume the flow computation for $5\,T_{\text{shedd}}$ seconds to generate the snapshots.
- In the Graphics and Animation task page, use Solution Animation Playback to animate the iso-vorticity lines, and save the results as a video-clip.

Question 11. *Generate a video-clip of the vortex shedding, and insert it in your* Word *report.*

the time interval (the incertitude is $\pm\frac{1}{2}\frac{1}{\Delta t_{FFT}}$ s^{-1} for an interval of length Δt_{FFT} s). So choose a large time interval for your FFT to have accurate results !