

Initiation à l'utilisation du logiciel de simulation numérique ANSYS/FLUENT

olivier.boiron@centrale-marseille.fr

Septembre 2019

3A FETES



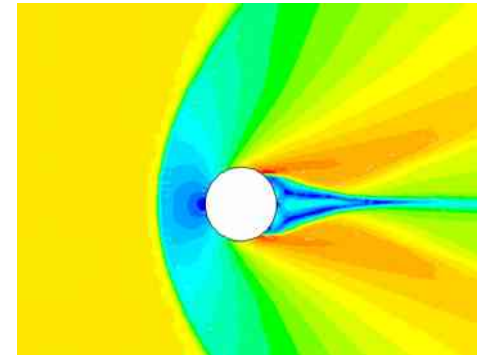
Introduction à l'utilisation du logiciel de simulation numérique ANSYS/FLUENT

- Introduction
- Exemples 2D
 - Ecoulements (laminaire et turbulent) autour d'un cylindre
 - Instabilité de Rayleigh-Bénard
 - Ecoulement diphasique (Euler/Lagrange)
- Exemple 3D
 - Ecoulement en conduite
- Fonctionnalités avancées
 - Maillage dynamique
 - Utilisation de journaux et calculs « batch »
 - Personnalisation – « user functions »

Introduction

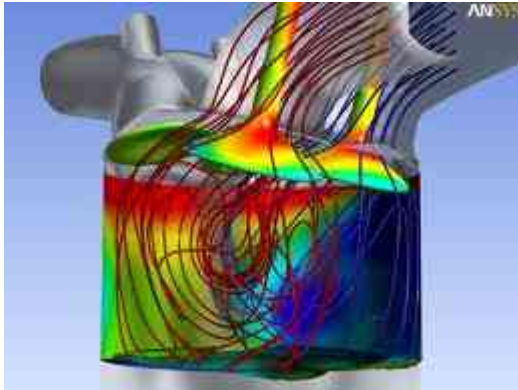
- Simulation numérique en mécanique des fluides : CFD

Computational Fluid Dynamic



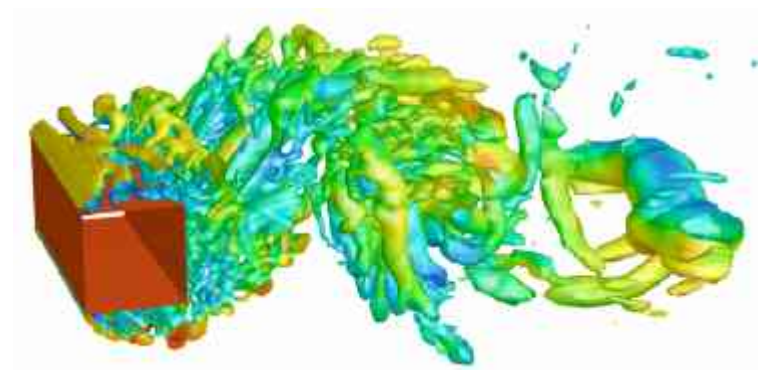
- Logiciels dédiés depuis le début des années 90 avec le développement des moyens de calculs et de méthodes performantes de calcul scientifique
- Tendance actuelle au multi-physique c ad au d veloppement de plateformes permettant d'analyser des probl mes coupl s de m canique, de thermique, de fluides, etc.
- ANSYS, COMSOL, Abaqus, Solidworks, **OpenFOAM**, DynaFlow, **GERRIS**, etc...

Introduction



La CFD est particulièrement utile pour :

- Pré-dimensionnement
 - Optimisation de forme/process
 - Calcul/Dimensionnement d'efforts sur des structures
 - Visualisation des champs de pression, vitesse, etc
- Calcul de physiques complexes :
 - Ecoulements turbulents
 - Ecoulements Compressibles
 - Ecoulements réactifs - Combustion
 - Ecoulements polyphasiques
 - Ecoulements à surface libre



Introduction

Implémentation d'un problème de CFD / Méthodologie

Pour faire de la CFD il est nécessaire d'avoir :

- Une formulation mathématique du problème (EDP+CL+Domaine de calcul)
 - Par exemple équations d'Euler, Navier-Stokes, RANS,...
- Une méthode numérique permettant de résoudre le problème
 - Différences finies
 - Eléments finis
 - **Volumes finis**
- Une représentation discrète ou maillage du domaine de calcul
- Un calculateur

Introduction

Implémentation d'un problème de CFD / [Méthodologie](#)

Méthodologie en [5 étapes](#)



- Analyse du problème : modèles physiques, géométrie, conditions aux limites, propriétés physiques,...
- Création de la géométrie avec un [modeleur](#)
- Création du maillage avec un [mailleur](#)
- Définition des données du problème (physiques, CL, propriétés physiques, schémas numériques,..) et résolution avec le [solveur](#)
- Exploitation des résultats avec le [post-processeur](#)

Introduction

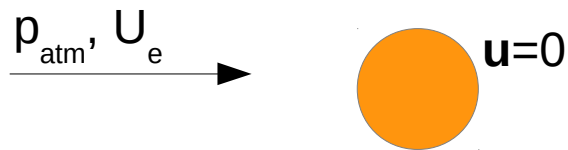
Implémentation d'un problème de CFD / Analyse

- Identification de la physique :
 - Modèle physique : équations, propriétés physiques, ...
 - Quelles sont les variables inconnues ? ..., ρ , \mathbf{u} , T , μ , k , ε ,
- Conditions aux limites :
 - Identifier les conditions aux limites et leur type (Dirichlet, Neuman)
 - Mettre en correspondance les CL avec les traitements proposés aux frontières par le solveur
- Domaine de calcul
 - Définir le domaine de calcul, *i.e.*, l'espace physique dans lequel les équations seront résolues.
 - Sauf cas particulier les représentations informatiques de domaine de calcul sont des espaces finis. Comment prendre en compte cette contrainte ?
 - Existence de symétries ?

Introduction

Implémentation d'un problème de CFD / Analyse

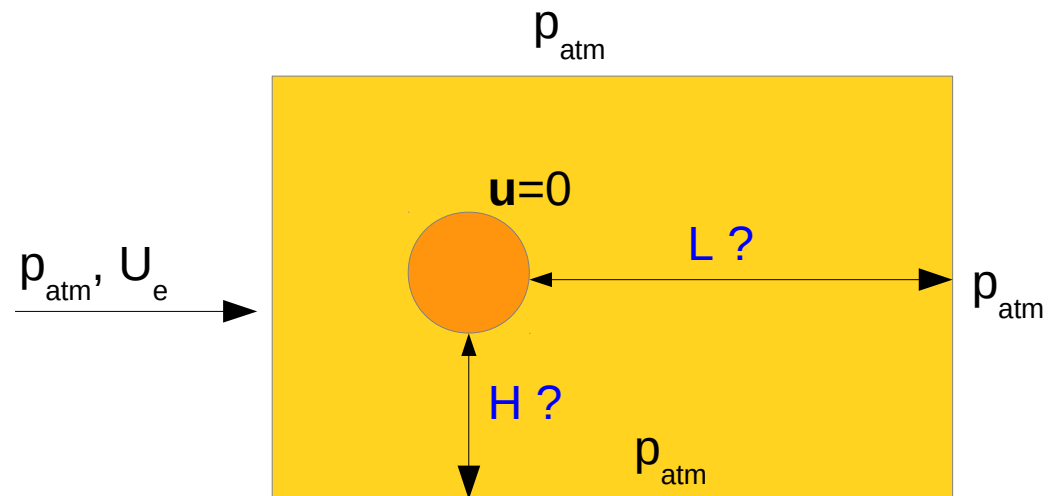
Exemple : **écoulement externe** autour d'un objet.



- A l'amont la vitesse est imposée
- Sur l'objet, adhérence du fluide
- « **Loin** » de l'objet $u=U_e$

Transposition à un domaine fini

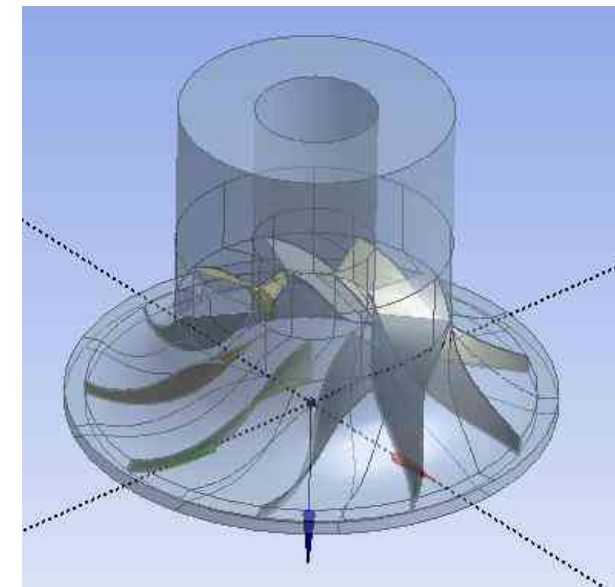
- Choix non unique pour les CL:
 - Conditions de symétrie
 - Pression imposée
- Quelles dimensions géométriques ?



Introduction

Implémentation d'un problème de CFD / Modelage

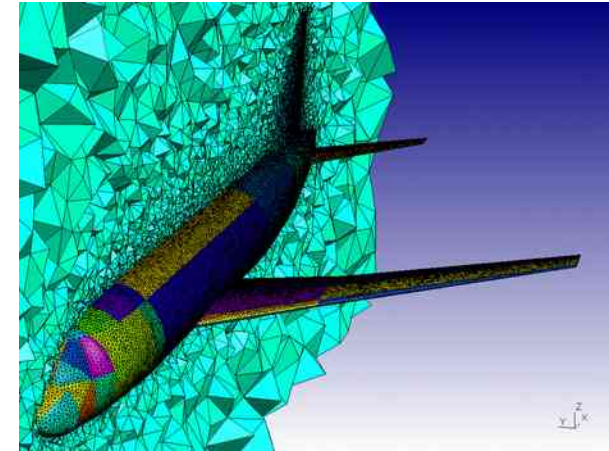
- Modelage= Représentation du domaine de calcul
- On représente le domaine ou espace dans le lequel le fluide circule et non uniquement l'objet autour duquel il circule !!
- Identifier les zones utiles pour définir correctement les conditions aux limites et créer les entités géométriques *ad hoc*
- Identifier les zones utiles pour établir des post traitements particuliers : par exemple résultante des efforts exercés sur une surface donnée d'un corps → il est habituellement plus simple de définir cette surface dès la conception du domaine plutôt que de le faire avec le post processeur
- Import direct de géométrie en provenance de logiciel de CAO possible → Résultats très dépendants du format natif (par exemple iges généralement à éviter!)



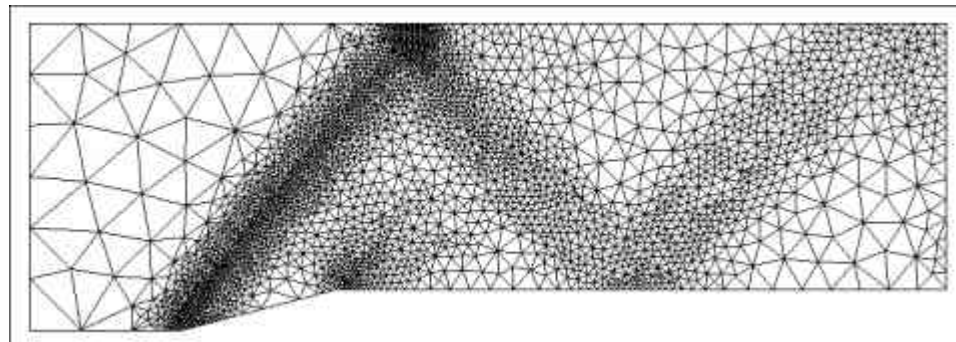
Introduction

Implémentation d'un problème de CFD / Maillage

- Création du maillage avec un **mailleur** !!!
- **Etape généralement la plus consommatrice en temps !!!!**
- Optimisation nécessaire :
 - Taille des mailles \longrightarrow Précision calcul
 - Taille du maillage \longrightarrow Temps de calcul



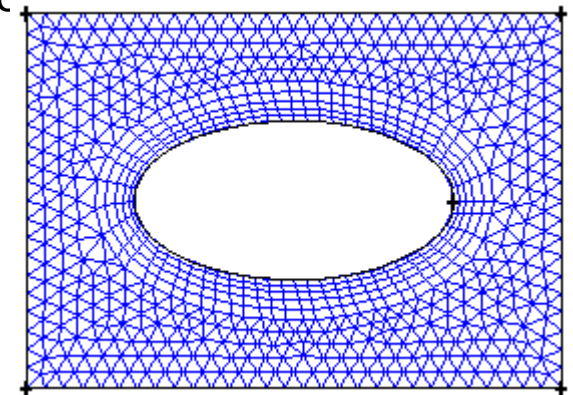
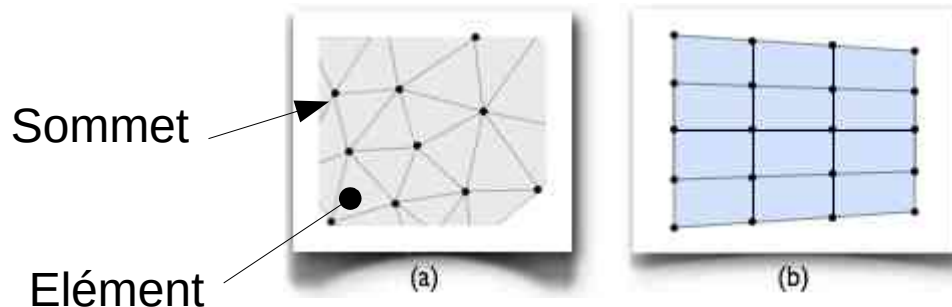
- La taille des mailles doit varier de manière progressive
- La densité des éléments en certains endroits du domaine doit être plus importante en cas de forts gradients
- La précision des schémas numériques est proportionnelle à :
 - Δx pour un schéma de discrétisation au 1^{er} ordre
 - $(\Delta x)^2$ pour un schéma de discrétisation au 2^{ème} ordre, etc



Introduction

Implémentation d'un problème de CFD / Maillage

- Le maillage est un ensemble recouvrant « au mieux » le domaine géométrique étudié
- Il peut être constitué d'éléments linéiques, surfaciques et volumiques
- La **connectivité d'un maillage** est la définition des liaisons entre les sommets de ses éléments
- On parle de **maillage hybride** lorsqu'il mêle des éléments de nature différente. Par exemple en 2D quadrangle et triangle ou en 3D hexaèdres et tétraèdres
- Le maillage est qualifié de **structuré** (b) lorsque la connectivité des nœuds est de type différences finies
- Un maillage est dit **non structuré** (a) lorsque la connectivité est quelconque

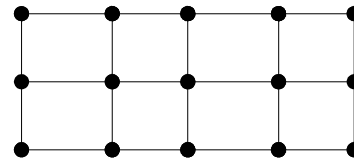


Maillage hybride non structuré

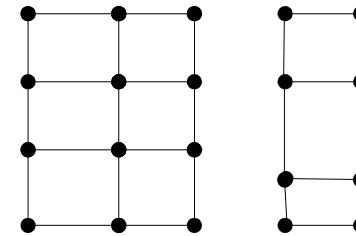
Introduction

Implémentation d'un problème de CFD / Maillage

- Le maillage M d'un domaine Ω est dit **conforme** si :
 - $\tilde{\Omega} = \bigcup_{K \in M} K \approx \Omega$
 - Tout élément (ou cellule) K de M est d'intérieur non vide
 - L'intersection de deux éléments de M doit être soit :
 - \emptyset
 - Un point
 - Une arête
 - Une surface

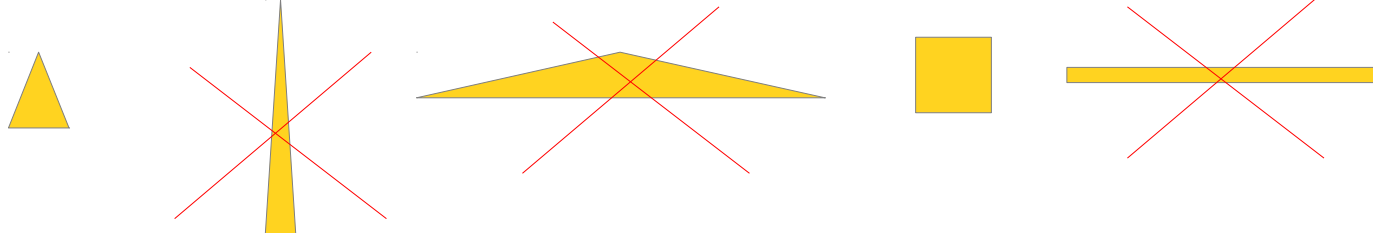


Conforme



Non conforme

- Les éléments ne doivent pas être trop déformés ou autrement dit être les plus réguliers possible



Introduction

Implémentation d'un problème de CFD / Maillage

- Création du maillage avec un [mailleur](#)
- Outils inclus dans le mailleur pour s'assurer de la qualité du maillage : par exemple le critère [EQUIANGLE SKEW](#) :

$$Q_{eq} = \max\left(\frac{\theta_{max} - \theta_{eq}}{180 - \theta_{eq}}, \frac{\theta_{eq} - \theta_{min}}{\theta_{eq}}\right)$$

Θ_{max} = est l'angle max sur une face ou une cellule

Θ_{min} = est l'angle min pour une face ou une cellule

$\Theta_{eq} = 60^\circ$ pour un triangle ou un tétraèdre équilatéraux

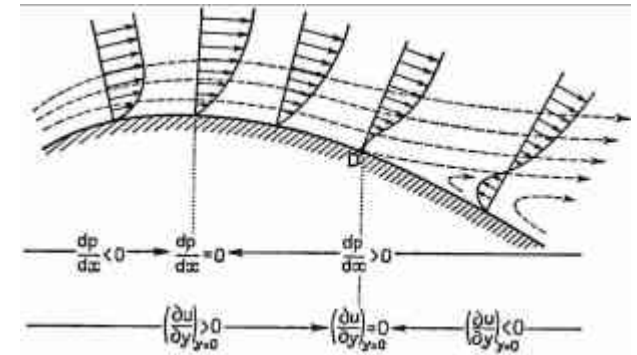
$\Theta_{eq} = 90^\circ$ pour un quadrilatère ou un hexaèdre équilatéraux

Des valeurs de Q_{eq} au delà de 0.6 caractérisent des [maillages problématiques](#) !!!!

Introduction

Implémentation d'un problème de CFD / Maillage

- Problématique de la taille des mailles au voisinage des parois
- En fluide visqueux présence de **couches limites** sur les parois.



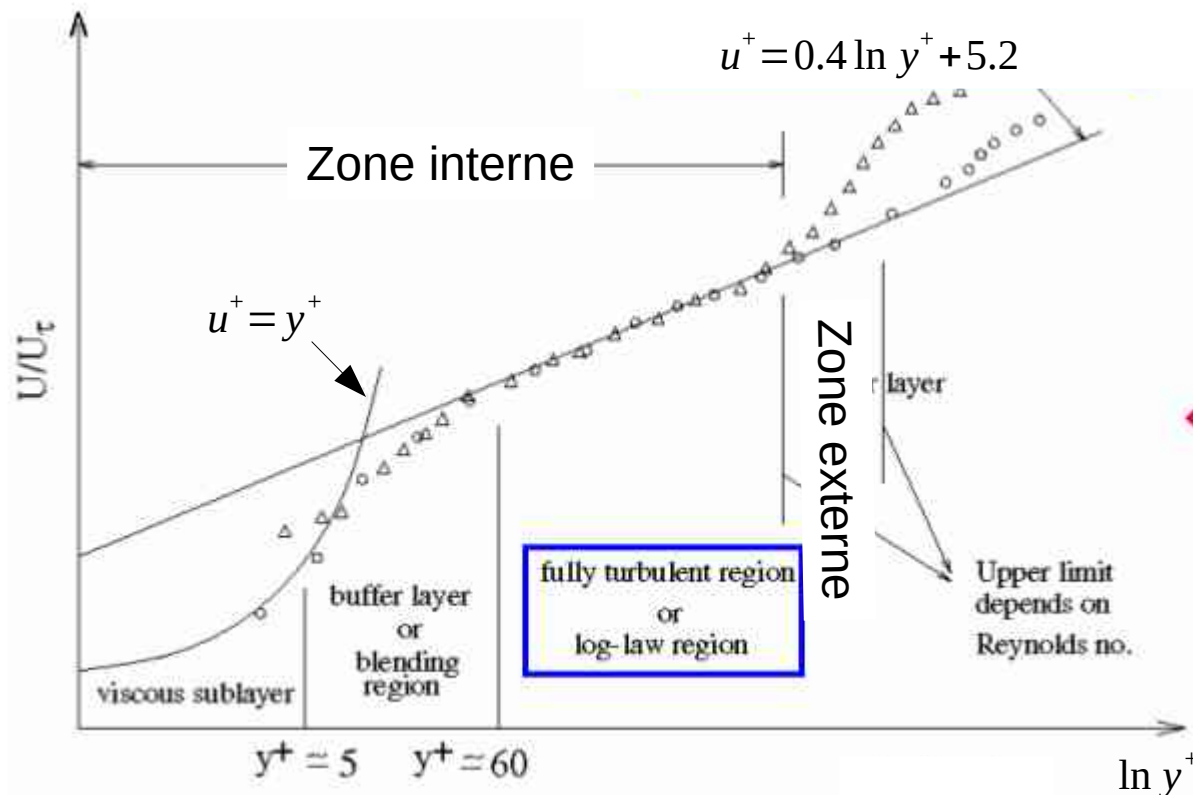
Couche limite turbulente

$$U^* = \sqrt{\frac{\tau_p}{\rho}} \quad \text{vitesse de frottement}$$

$$y^+ = \frac{y \cdot U^*}{\nu} \quad \text{coordonnées réduite}$$

$$u^+ = \frac{\bar{u}}{u_*} \quad \text{vitesse réduite}$$

$$C_f = \frac{\tau_p}{0,5 \rho U_e^2} \quad \text{coeff de frottement}$$



Introduction

Implémentation d'un problème de CFD / Maillage

- Nécessité de raffiner le maillage pour intercepter correctement les gradients de vitesse et le frottement à la paroi τ_p
- Estimation de l'épaisseur de la couche limite avec les lois classiques de plaques planes:

- En laminaire ($Re_L < 5 \cdot 10^5$):

$$\frac{C_f}{2} = 0,664 Re_L^{-\frac{1}{2}}$$

- En turbulent ($Re_L < 4 \cdot 10^6$):

$$Re_L = \frac{\rho U_e L}{\mu} \quad \frac{\bar{C}_f}{2} = 0,037 Re_L^{-\frac{1}{5}}$$

L est la longueur de la paroi/plaque
U_e vitesse extérieure de l'écoulement.

Exemple pour de l'air: U_e = 20 m/s – ρ = 1,2 kg/m³ – μ = 1,75 · 10⁻⁵ kg/m/s - L = 1m

$$Re_L = 1,37 \cdot 10^6 \rightarrow \bar{C}_f = 0,0044$$

$$U_* = \sqrt{\frac{\bar{C}_f U_e^2}{2}} = 0,94 \text{ m/s}$$

$$y_{y^+=1} = \frac{\mu}{\rho U_e} \approx 1,55 \cdot 10^{-5} \text{ m}$$

Introduction

Implémentation d'un problème de CFD / Maillage

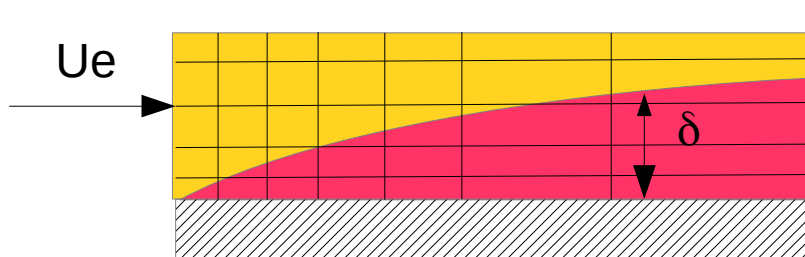
- En pratique en laminaire on essaye de placer au minimum 3 mailles / cellules dans la couche limite.
- En turbulent la situation est plus complexe car dans la couche limite turbulente (CLT) la zone de proche paroi (sous couche visqueuse) est très mince. Deux cas de figures :
 - Le solveur utilise des lois de parois pour modéliser l'écoulement en proche paroi
 - Les lois de paroi sous couche visqueuse, zone tampon et zone log sont universelles. Il suffit alors de placer une maille dans la zone log c'est-à-dire $y^+ > 50$ et la vitesse dans la zone située en deçà de cette valeur est calculée par les lois de la CLT.

Introduction

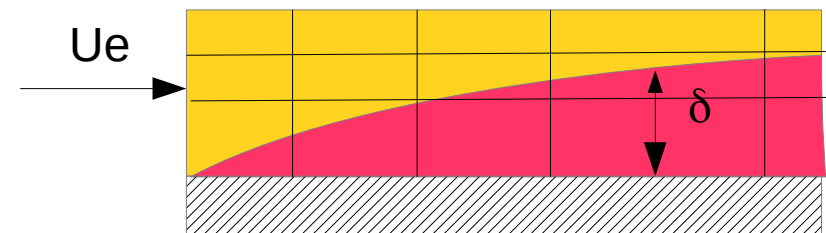
Implémentation d'un problème de CFD / Maillage

- Le solveur n'utilise pas des lois de parois :
- Il faut mailler jusqu'à la sous couche visqueuse, $y^+ < 1$. Ceci s'avère très contraignant car les mailles sont d'autant plus fines que le Re est grand....
- Dans la sous couche visqueuse on a $u^+ = y^+$ avec : $y^+ = \frac{y u^*}{\nu}$

et : $u^* = \sqrt{\frac{\tau_p}{\rho}}$ la vitesse de frottement.



Sans loi de paroi
1ère maille dans
la s.c.v. ($y^+ < 1$)

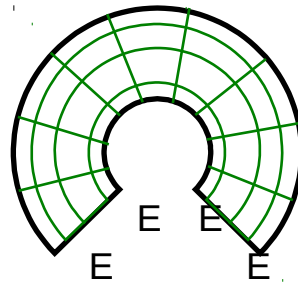
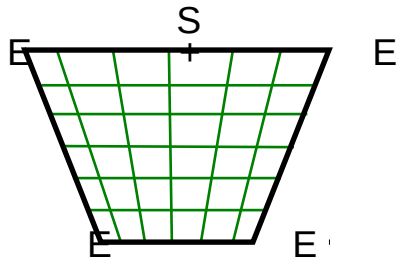


Avec loi de paroi :
1ère maille dans
la zone log. ($y^+ > 50$)

Introduction

Implémentation d'un problème de CFD / Maillage

- Divers algorithmes de maillages sous GAMBIT. Par exemple en 2D :
 - Map ou Submap
 - Maille des domaines topologiquement équivalent à des quadrangles ou des ensembles de quadrangles
 - Nécessite de définir la densité de mailles sur les frontières des domaines
 - Possibilité de définir la densité de mailles sur les frontières

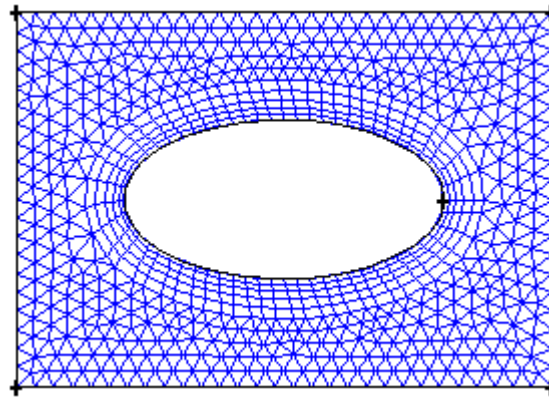


Un quadrangle possède 4 sommets de type **END (E)**

Introduction

Implémentation d'un problème de CFD / Maillage

- Divers algorithmes de maillages sous GAMBIT. Par exemple en 2D :
 - PAVE
 - Maille des domaines quelconques particulièrement domaine non convexe
 - Nécessite de définir la densité de mailles sur les frontières des domaines
 - Contrôle de la densité de maille par l'intermédiaire de « fonctions de mesure »



Introduction

Implémentation d'un problème de CFD / Maillage

- On se doit de vérifier **la sensibilité des résultats de la simulation au maillage**
- Calculs sur plusieurs maillages (2 ou 3) de densité différentes pour vérifier l'insensibilité des résultats à la finesse du maillage → **Convergence au maillage**
- On choisit des critères de comparaison :
 - Locaux : par exemple position du point de décollement, de rattachement, valeurs de champs en des points particuliers, etc
 - Globaux : calcul de coeff de frottement, de traînée, d'échange de chaleur, etc...

Implémentation d'un problème de CFD / Solveur

- La **méthode des Volumes Finis** est la méthode de résolution des équations de la mécanique des fluides utilisée par FLUENT
- L'idée générique de la méthode est de transposer à un problème continu un **problème discret** permettant de déterminer la valeur des champs inconnus en des points particuliers pré-déterminés dits « **centre de cellule** »
- Ainsi, très schématiquement, le problème continu constitué par le triptyque EDP+CL+Domaine de calcul est remplacé par un système d'équations algébriques linéaire de la forme :

$$A \cdot X = B$$

- X est le vecteur inconnu : $X = \left[p_0 \ p_1 \dots p_{n-1} \ u_0 \ u_1 \dots u_{n-1} \ v_0 \ v_1 \dots v_{n-1} \right]^T$
- n est le nombre de cellules ou « taille du maillage » contenues dans le domaine de calcul
- p_i , u_i et v_i respectivement la valeur de la pression et des deux composantes de la vitesse en ces points (seules inconnues pour un pb 2D incompressible et isotherme)
- B est une matrice nxn, creuse, dont la forme précise dépend, entre autres des schémas de discrétisation retenus.

Implémentation d'un problème de CFD / Solveur

- FLUENT résout le problème précédent en utilisant une **méthode de Gauss-Seidel**
- On définit un vecteur initial x^0 , le plus souvent le vecteur nul, et l'algorithme calcule la suite :

$$x^0, X^1, \dots, X^k$$
$$\left[A \cdot X^k - B \right] < \epsilon$$

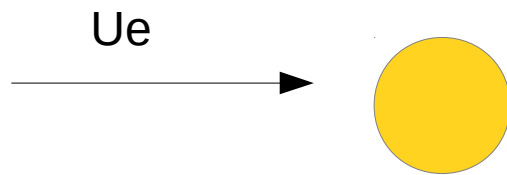
- Le calcul est arrêté lorsque : où ϵ est le résidu
- En fait dans FLUENT la convergence de chaque variable inconnue est testée. La valeur par défaut du résidu est fixée à 10^{-3} .
- Si l'algorithme ne converge pas il s'arrête automatiquement lorsque le nombre maximum d'itérations défini par l'utilisateur, $n_{\text{iter_max}}$, est atteint
- En pratique, la taille des maillages étant très importante ($> 10^5$ éléments) FLUENT met en œuvre une résolution sur des maillages plus grossiers autogénérés et imbriqués successivement jusqu'au maillage final. Cette technologie porte le nom de « **multigrille** » et se décline dans FLUENT en deux algorithmes **Algebraic MultiGrid** ou « **AMG** » et **Full Approximation Storage** ou « **FAS** » pour les problèmes fortement compressibles

Implémentation d'un problème de CFD / Solveur Couplage pression-vitesse

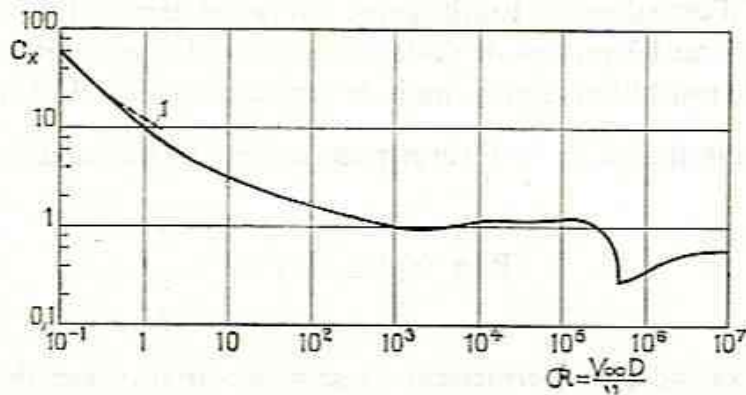
- Lorsque le fluide est incompressible le problème est que le terme source principal de l'équation de quantité de mouvement est la pression et que celle-ci n'est pas déterminée implicitement par une équation propre.
- Il s'en suit que le champ de vitesse calculé par résolution de l'équation de QDM ne vérifie pas l'équation de continuité !!!
- Plusieurs possibilités par le biais des algorithmes de couplage pression-vitesse. Par exemple l'algorithme SIMPLE pour Semi Implicit Method for Pressure Linked Equations (Spalding & Patankar, 1972)
 - Imposition des conditions aux limites
 - Calcul des gradients et flux aux interfaces
 - Résolution de l'équation discrétisée de QDM
 - Calcul des erreurs de flux aux interfaces
 - Résolution de l'équation de correction de la pression
 - Fixation des nouvelles valeurs de la pression aux nœuds du maillage
 - Correction de la valeur du champ de vitesse

Plusieurs algorithmes sont proposés par Fluent. **Par défaut l' algorithme SIMPLE est utilisé**

Écoulement autour d'un cylindre 2D



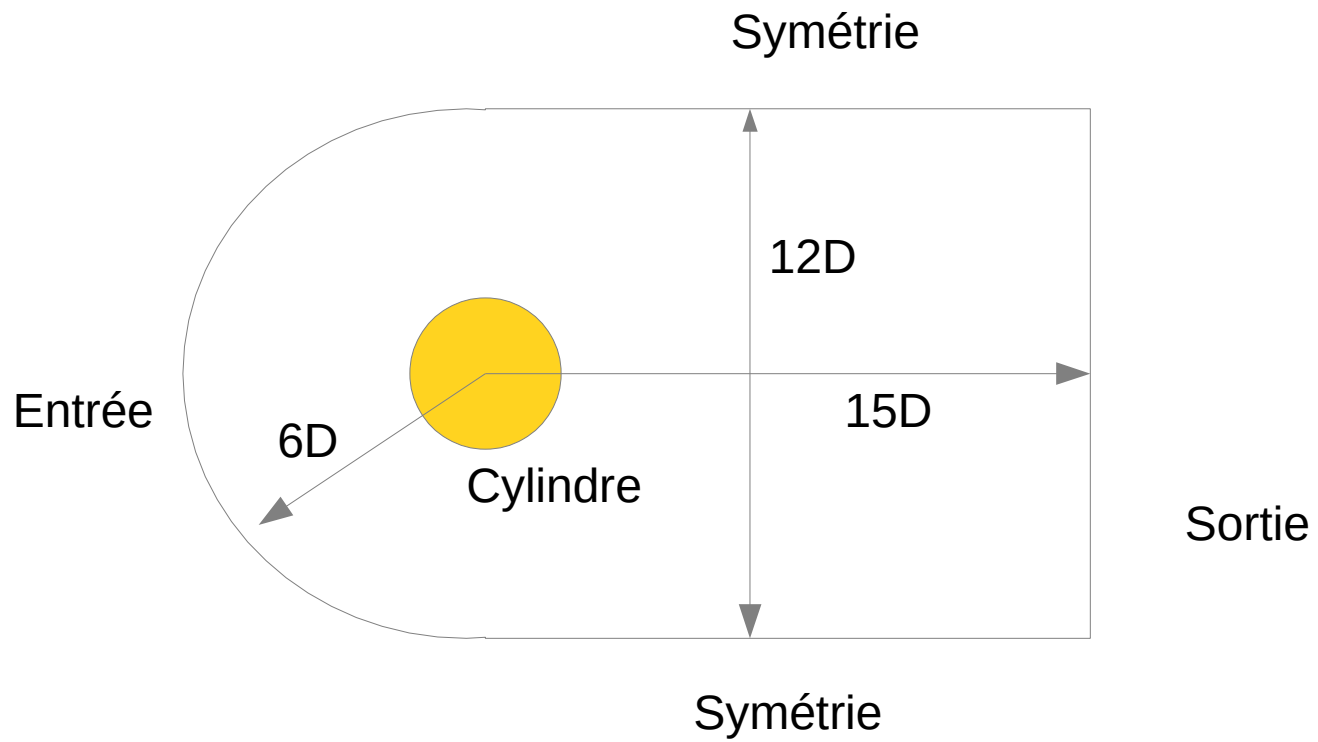
- Efforts exercés sur le cylindre ?
- Champs de vitesse et de pression ?
- Allure du sillage ?



5 configurations d'écoulements :

- Laminaire $Re_D = 1$
- Laminaire $Re_D = 50$
- Laminaire $Re_D = 100$
- Turbulent $Re_D = 50\,000$
- Compressible $M=2$

Écoulement autour d'un cylindre 2D



Écoulement autour d'un cylindre 2D

Hypothèses : 2D plan
Stationnaire
Incompressible
Fluide newtonien

Conservation de la masse

$$\nabla \cdot u = 0$$

Conservation de la quantité de mouvement

$$(u \cdot \nabla) u = \frac{1}{\rho} \nabla \cdot \sigma$$

Loi de comportement rhéologique du fluide

$$\sigma = -pI + 2\mu D$$

Tenseur des vitesses de déformation

$$D = \frac{1}{2} (\nabla u + \nabla u^T)$$

2 inconnues $u \begin{vmatrix} u \\ v \end{vmatrix}$ et p

Le système d'EDP à résoudre est non linéaire (termes convectifs)
et d'ordre 2 en espace

Ecoulement autour d'un cylindre 2D

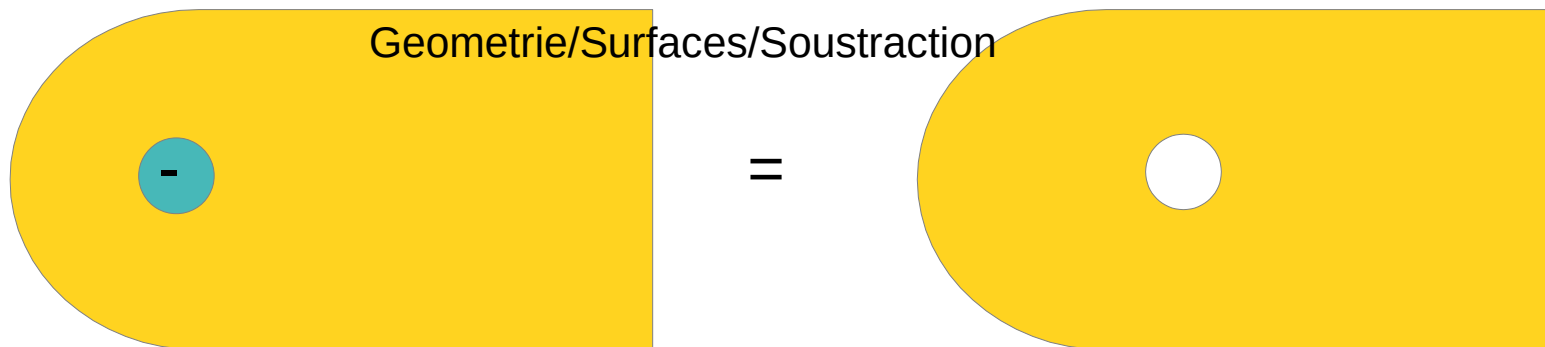
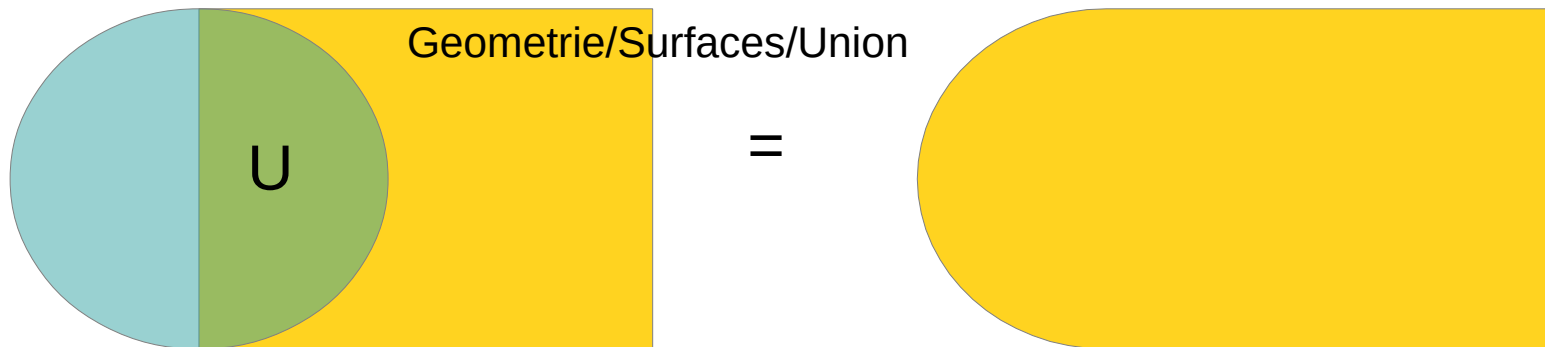
Les conditions aux frontières:

Correspondance CL Fluent

- A l'entrée du domaine: $u = U_0$ → Velocity inlet
- Sur le cylindre: $u = 0$ → Wall
- A la sortie du domaine: $p = cte$ → Pressure outlet
- Sur le haut et sur le bas: $\left\{ \begin{array}{l} u \cdot n = 0 \\ \frac{\partial u_\tau}{\partial n} = 0 \end{array} \right.$ → Symetry

Écoulement autour d'un cylindre 2D

Construction de la géométrie avec GAMBIT (obsoleète):



L'échelle du maillage est ajustable dans FLUENT: on peut travailler ici en *cm*

Écoulement autour d'un cylindre 2D

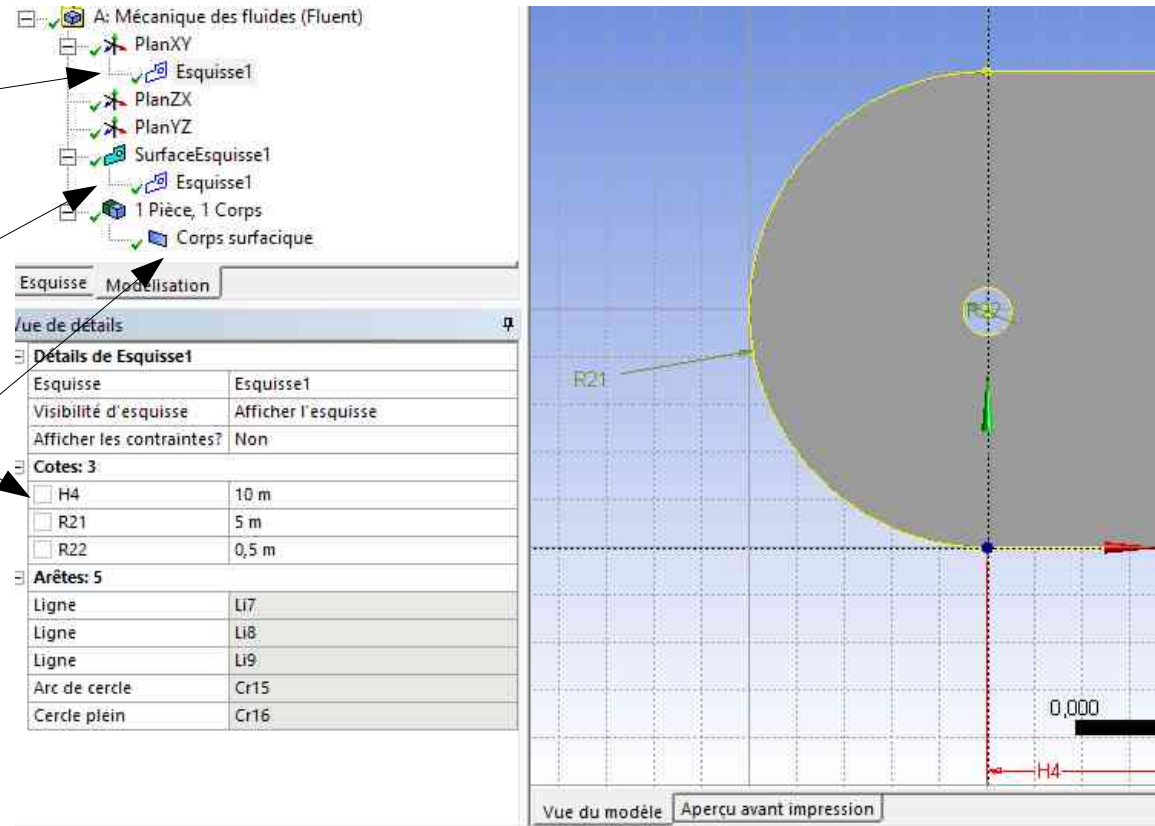
Construction de la géométrie avec Design Modeler:

Travail en mode esquisse pour définir les contours

Dimensions définies par des côtes

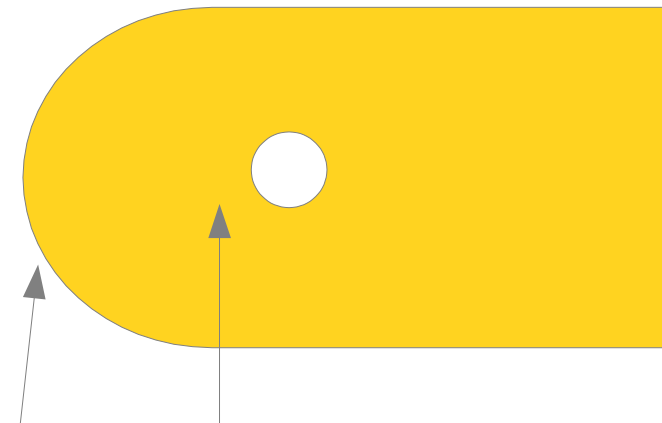
Surfaces définie à partir de l'esquisse

Pièce pour définir l'entité



Écoulement autour d'un cylindre 2D

Construction du maillage avec GAMBIT (obsoleter):



- Emploi d'une *size function* pour définir le raffinement.
- Principe: construction d'une suite géométrique définissant la taille des éléments lorsque l'on s'éloigne dans la direction normale à une entité géométrique source

source: c'est l'entité à partir de laquelle la distance est définie
ici l'arête qui définit le contour du cylindre

attachment: c'est l'entité qui sera maillée, ici la surface en bleue

start size: distance de la première maille à la paroi: 0.0314

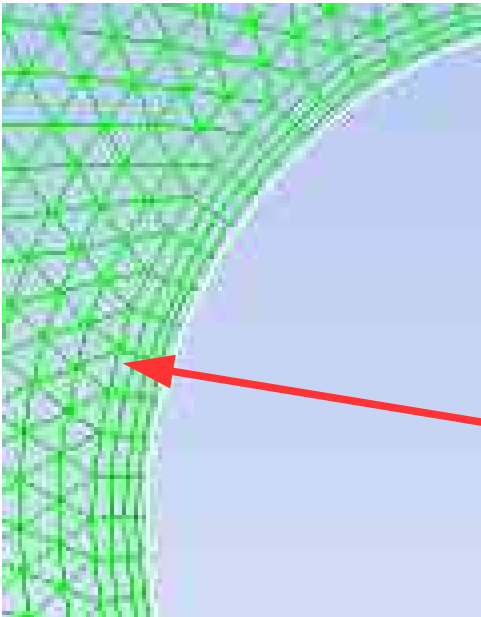
growth rate: raison de la suite géométrique: 1.1

max size: taille limite du maillage: 0.314

Écoulement autour d'un cylindre 2D

Construction d'un maillage limitant la diffusion numérique

- Discrétisation par schéma décentré amont plus stable mais source de diffusion numérique
- Augmenter l'ordre des schémas améliore la précision mais pose des pb de convergence



Pour y remédier :

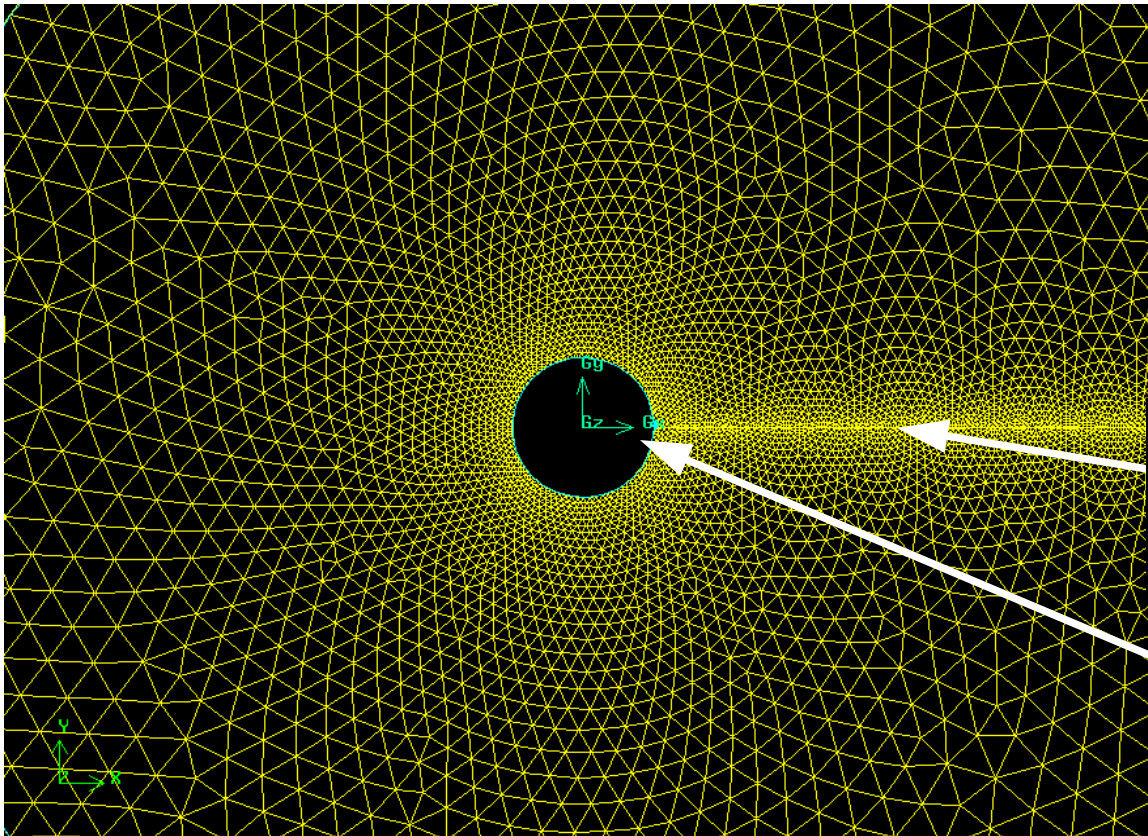
- Raffinement
- Alignement des mailles sur la direction de l'écoulement

Maillage hybride :

Raffiné sur le cylindre avec quadrangle
Raffiné sur la ligne moyenne du sillage

Ecoulement autour d'un cylindre 2D

Construction du 1^{er} maillage avec GAMBIT:



Elément triangle
Raffiné sur le cylindre
Raffiné sur la ligne moyenne du sillage

start size= 0.0314
growth rate =1.2
max size = 0.314

start size= 0.0314
growth rate =1.15
max size = 0.314

Écoulement autour d'un cylindre 2D

Construction du 1^{er} maillage avec Mesher:

Méthode Triangles

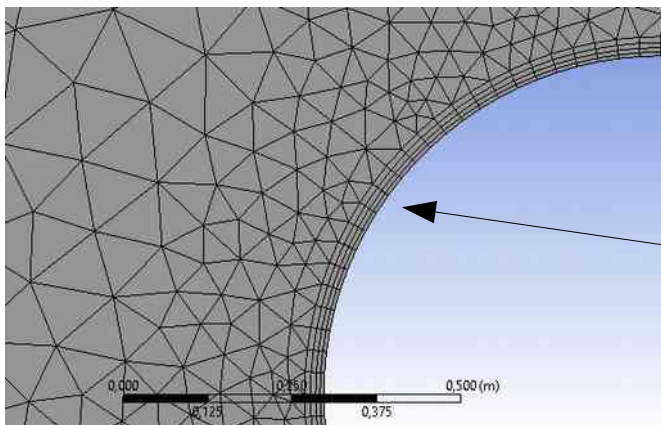
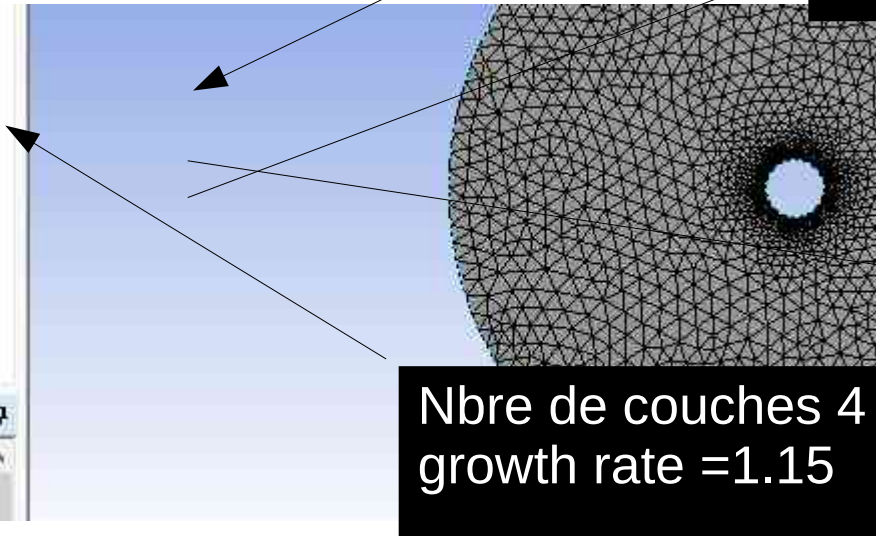
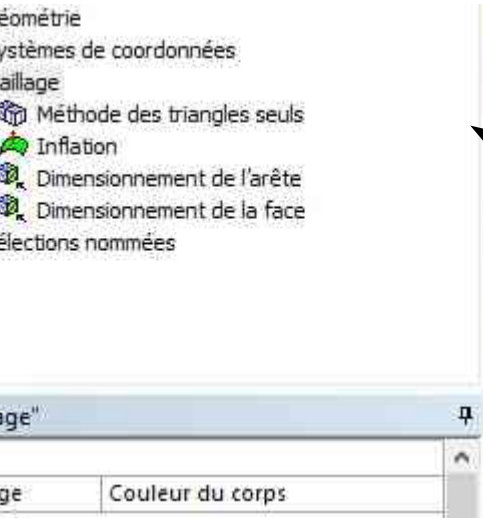
Taille éléments 0.314

Taille éléments 0.0314

Nbre de couches 4
growth rate =1.15

6736 Eléments de type triangle/quadrangle
Raffiné sur le cylindre

Maillage hybride



Écoulement autour d'un cylindre 2D

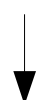
Les fichiers importants:

Maillageur

mon_cas.jou



mon_cas.dbs



mon_cas.msh

Fluent

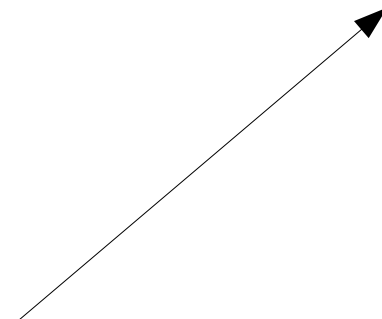
mon_cas.cas.gz



mon_cas.dat.gz

mon_script.txt

Format zip pour
gagner de la place !!!



- Les scripts permettent :
 - d'automatiser le traitement
 - Effectuer des calculs paramétriques

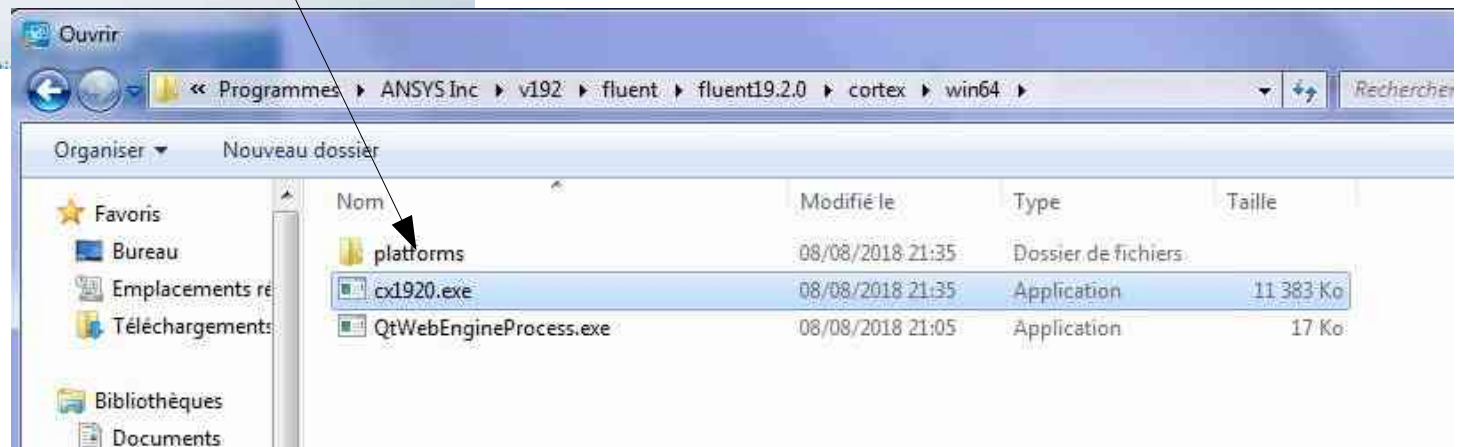


Fluent, à partir de la version 18.2 ne tourne pas sous Windows sous carte graphique intel.....

Il faut désactiver dans les paramètres 3D graphiques (clic droit dans le fond d'écran) l'antialiasing



Ajouter l'application **cx1920.exe** dans les applis 3D à réglages spécifiques





Fluent, à partir de la version 18.2 ne tourne pas sous Windows sous carte graphique intel.....

Désactiver l'anti-aliasing pour cette application.....

A screenshot of the Intel Graphics Control Panel interface. The window title is partially visible as 'ati...'. The Intel logo is in the top right corner. The main content is divided into two columns. The left column has a section titled 'Sélectionner l'application' with a dropdown menu showing 'Applications avec paramètres 3D personnalisés' and '1920' selected. Below this are 'Parcourir' and 'Analyser' buttons. The right column shows settings for '1920'. Under 'Anticrénelage morphologique conservateur', the 'Éteindre' option is checked. Under 'Anticrénelage multi-échantillon', the 'Éteindre' option is also checked. Other settings like 'Synchronisation verticale' and 'Filtrage anisotrope' are visible but not fully detailed. Arrows from the text above point to the 'Éteindre' checkboxes.

Ecoulement autour d'un cylindre 2D

Paramétrage du solveur

- **File/Read Case/Mesh** : lecture maillage
- **Grid/Scale** : passage en SI si la CAO n'est pas en unité SI
- **Define/Model** :
 - Dimension 2D, 2Daxi, 3D
 - Stationnaire ou Instationnaire
- **Define/Materials** : sélection du/des fluide(s) utilisé(s)
- **Define/Boundary Conditions** : paramétrage des CL
- **Solve/Controls** : paramétrage schémas numériques
- **Solve/Monitoring** : paramétrage monitoring & critères convergence
- **Solve/Initialization** : initialisation du vecteur inconnu
- **Solve/Iterate** : lancement calcul

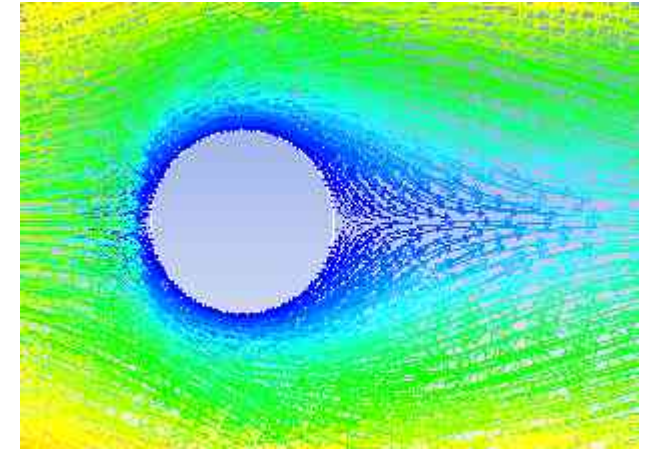
- **File/Write/Case et/ou Data**: sauvegarde fichiers cas et résultats

Écoulement autour d'un cylindre 2D

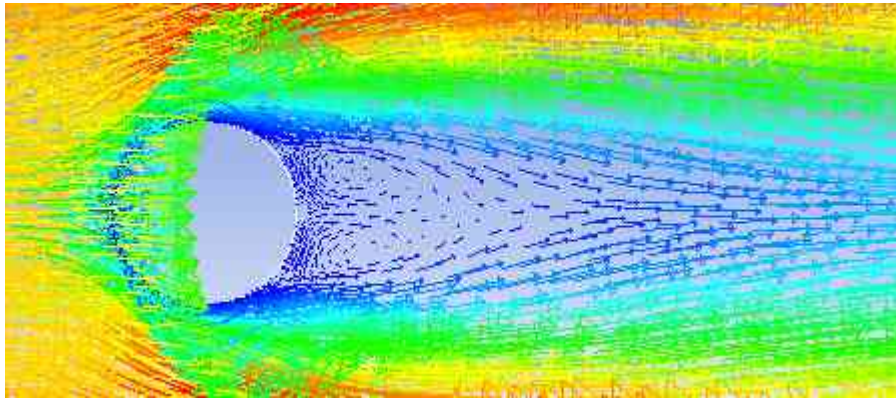
$$Re_D = \frac{\rho U D}{\mu} = 1$$

Écoulement rampant ou de Stokes

- Les effets visqueux sont prépondérants
- L'écoulement est symétrique i.e. les lignes de courant sont identiques si l'on renverse l'écoulement de 180°
- Pas de décollement à l'arrière du cylindre



$$Re_D = \frac{\rho U D}{\mu} = 20$$



Écoulement décollé

- L'écoulement perd sa symétrie
- Présence d'un décollement
- Deux tourbillons contra-rotatifs apparaissent à l'arrière du corps
- Sillage + important

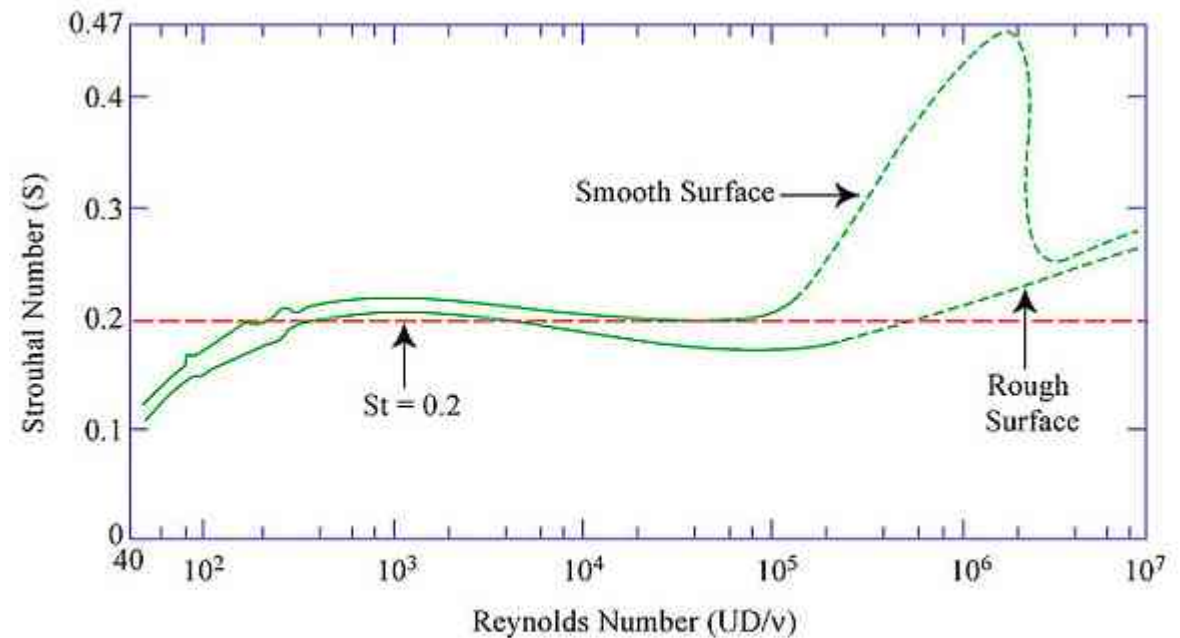
$$Re_D = \frac{\rho U D}{\mu} = 125$$

Écoulement autour d'un cylindre 2D

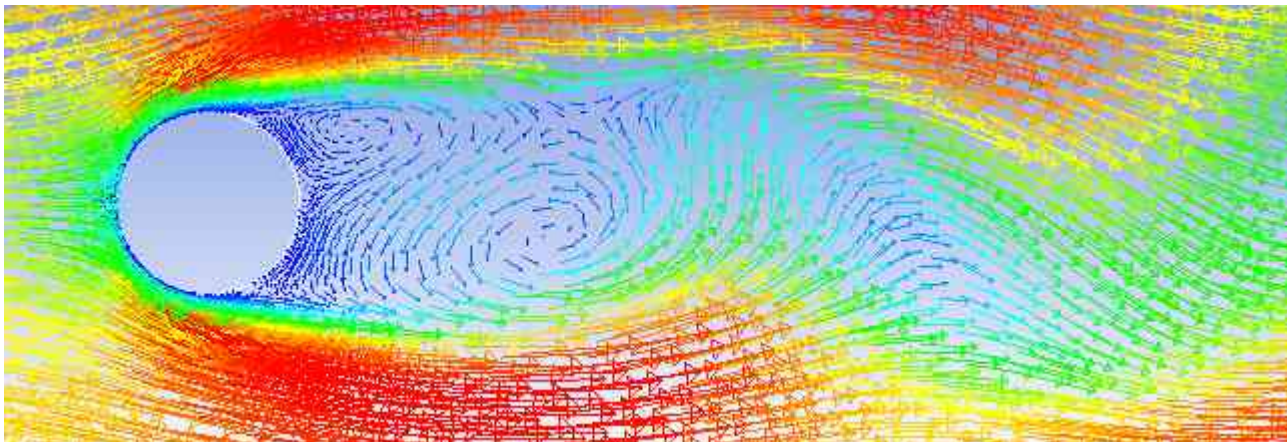
Allée de Von Karman

- Apparaît dès $Re \sim 25$
- L'écoulement est instationnaire
- Caractérisée par deux tourbillons contra-rotatifs qui se détachent de l'arrière du cylindre alternativement
- Fréquence du détachement caractérisée par le nombre de Strouhal :

$$St = \frac{Df}{U}$$



Relationship between Strouhal number and Reynolds number for circular cylinders. Data from Lienhard (1966) and Achenbach and Heinecke (1981). $S \sim 0.21 (1 - 21/Re)$ for $40 < Re < 200$, from Roshko (1955).



$$Re_D = \frac{\rho U D}{\mu} = 125$$

Écoulements turbulents

Quand un écoulement est-il turbulent ?

- En écoulement externe :

$Re_x > 500\,000$ le long d'une surface $Re_x = \frac{\rho U x}{\mu}$

$Re_L > 20\,000$ autour d'un obstacle $Re_L = \frac{\rho U L}{\mu}$

- En écoulement interne :

$Re_L > 2\,300$

- Convection naturelle :

$Gr > 10^9$ $Gr = \frac{\rho^2 C_p g \beta \theta L^3}{\mu k}$

Gr est le nombre de Grashoff

Écoulement autour d'un cylindre 2D

$$Re_D = \frac{\rho U D}{\mu} = 5 \cdot 10^4$$

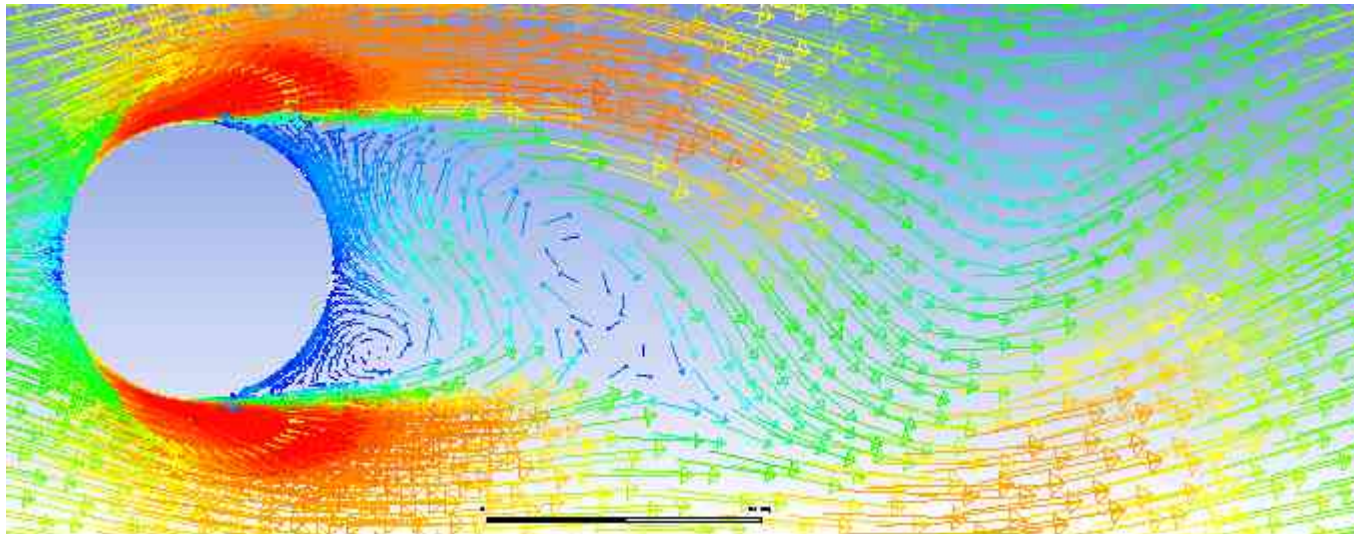
Écoulement turbulent

- Modèle k- ω SST (Shear Stress Transport)
- Initialisation du calcul en permanent puis passage en instationnaire
- Point de décollement plus en amont
- Le St est toujours voisin de 0.2

- 2 inconnues supplémentaires k et ω
- Il faut définir leur valeur sur les frontières notamment à l'entrée

$$I = 0.05 - l = 1 \text{ m}$$

- I est l'intensité de turbulence
- L l'échelle de longueur de la turbulence



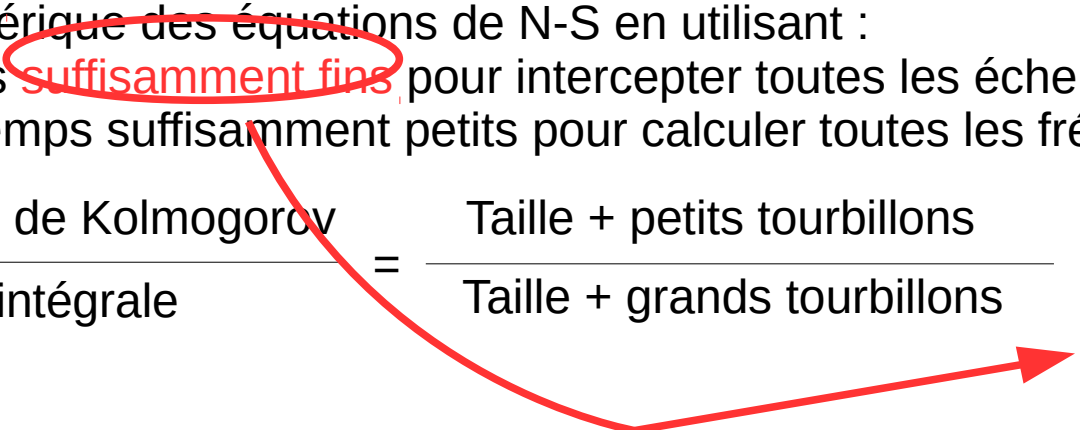
$$I = \frac{\sqrt{\overline{u'^2}}}{\bar{u}}$$

Écoulements turbulents

Les approches de modélisation de la turbulence

- Approche RANS – **Reynolds Averaged Navier-Stokes Equations**
 - Permet d'accéder aux propriétés moyennes de l'écoulement après **traitement statistique** (moyenne de Reynolds) des équations de N-S
 - Pas de calcul direct des diverses échelles de la turbulence qui sont modélisées
 - Très utilisé dans l'industrie
- Approche LES – **Large Eddy Simulation**
 - Calcul des grandes échelles de la turbulence uniquement
 - Les équations de N-S sont traitées par un **filtrage spatial** qui supprime les plus petites échelles
 - Les plus petites échelles sont modélisées par un « modèle de sous maille »
- Approche directe – **Direct simulation**
 - Résolution numérique des équations de N-S en utilisant :
 - Des maillages **suffisamment fins** pour intercepter toutes les échelles spatiales
 - Des pas de temps suffisamment petits pour calculer toutes les fréquences

Rappel :
$$\frac{\text{Micro-échelle de Kolmogorov}}{\text{Échelle intégrale}} = \frac{\text{Taille + petits tourbillons}}{\text{Taille + grands tourbillons}} = \frac{\eta}{l_{int}} = Re_{l_{int}}^{-\frac{3}{4}}$$



Écoulements turbulents

Equations RANS

- Application de la moyenne de Reynolds aux champs instantanés et décomposition en champ moyens et fluctuants.

Pour un champ scalaire $\xi(x,t)$ quelconque :

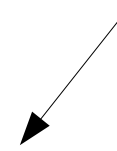
$$\xi(x,t) \quad \text{champ instantané}$$
$$\bar{\xi}(x,t) = \frac{1}{\Delta t} \int_{t-\Delta t}^{t+\Delta t} \xi(x,t) dt \quad \text{champ moyen au sens de Reynolds}$$
$$\xi'(x,t) = \xi(x,t) - \bar{\xi}(x,t) \quad \text{champ fluctuant}$$

- En substituant dans les équations de N-S (en incompressible) et en appliquant l'opérateur moyenne on obtient :

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0$$

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j} \left\{ \frac{\mu}{\rho} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \overline{u'_i u'_j} \right\}$$

Tenseur de Reynolds



Écoulements turbulents

Equations RANS

- Modélisation du tenseur de Reynolds par analogie avec le tenseur des contraintes visqueuses (Boussinesq)

$$\text{Tenseur de Reynolds} \quad -\rho \overline{u'_i u'_j} = \frac{\mu_t}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij}$$

- μ_t est la viscosité turbulente (ou tourbillonnaire)
- k est l'énergie cinétique de la turbulence $k = \frac{1}{2} \overline{u'_i u'_i}$
- μ_t est également modélisée par analogie avec la viscosité moléculaire (Boussinesq)

$$\mu_t = C_{te} \rho l_{int} u$$

- u est une échelle de vitesse caractéristique de la turbulence $u \approx k^{\frac{1}{2}}$
- l est une échelle de longueur caractéristique de la turbulence avec ϵ la dissipation visqueuse $l_{int} \approx \frac{k^{\frac{3}{2}}}{\epsilon}$

$$\mu_t = \rho C_{te} \frac{k^2}{\epsilon} \longrightarrow \text{Nécessite la connaissance de } k \text{ et } \epsilon \text{ pour être calculée !!}$$

Écoulements turbulents

Equations RANS – Modèle k-ε (Jones & Launder 1972)

$$\frac{\partial \bar{u}_i}{\partial x_i} = 0$$

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial x_j} \left\{ \frac{\mu + \mu_t}{\rho} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \right\}$$

$$\rho \left\{ \frac{\partial k}{\partial t} + \bar{u}_j \frac{\partial k}{\partial x_j} \right\} = \frac{\partial}{\partial x_j} \left\{ \left(\mu + \frac{\mu_t}{\sigma_k} \right) \frac{\partial k}{\partial x_j} \right\} + P - \rho \epsilon$$

$$\rho \left\{ \frac{\partial \epsilon}{\partial t} + \bar{u}_j \frac{\partial \epsilon}{\partial x_j} \right\} = \frac{\partial}{\partial x_j} \left\{ \left(\mu + \frac{\mu_t}{\sigma_\epsilon} \right) \frac{\partial \epsilon}{\partial x_j} \right\} + C_{\epsilon_1} \frac{\epsilon}{k} P - \rho C_{\epsilon_2} \frac{\epsilon^2}{k}$$

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon}$$

Inconnues :

\bar{u}_i, \bar{p}, k et ϵ

$$P = \frac{\partial \bar{u}_i}{\partial x_j} \frac{1}{2} \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right)$$

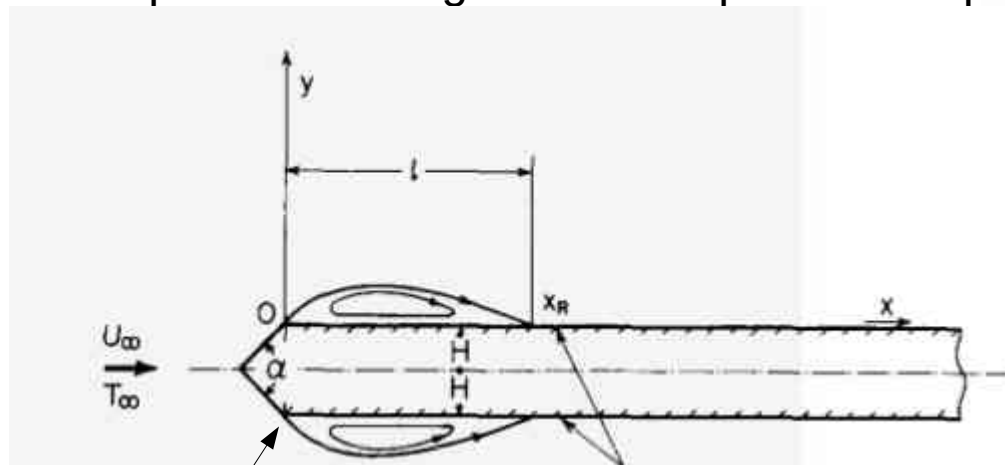
Production de k

Constantes (universelles) du modèle k-ε

C_μ	σ_k	σ_ϵ	C_{ϵ_1}	C_{ϵ_2}
0,09	1,00	1,30	1,44	1,92

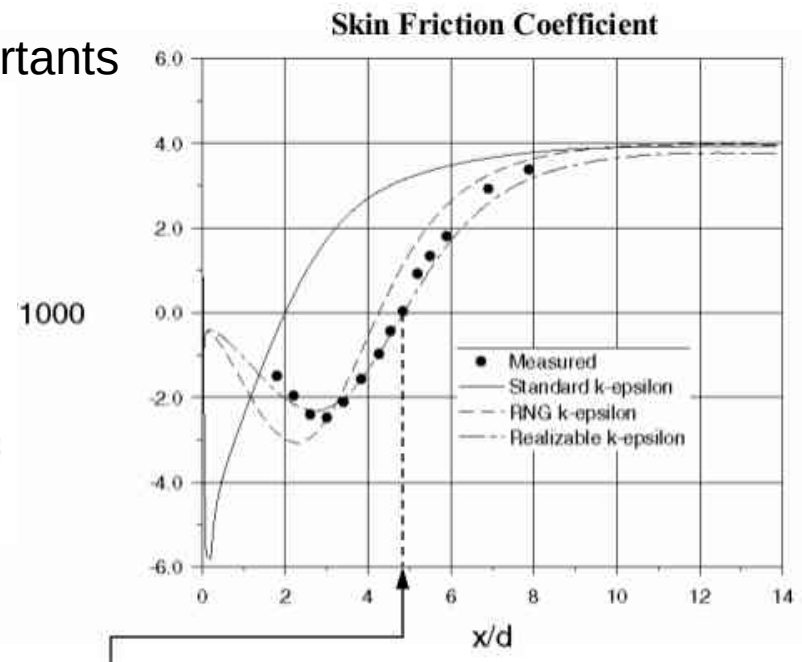
Écoulements turbulents

- Modèle $k-\epsilon$ est le premier modèle « universel »
- Très employé dans l'industrie car robuste mais souffre de nombreux défauts :
 - Calcul de l'équation de dissipation impossible à la paroi donc emploi de lois de parois obligatoire
 - Peu performant dans le cas d'écoulement avec décollement de CL
 - Peu performant si courbure importante
 - Peu performant si gradients de pression importants



Décollement

Point de rattachement expérimental pour $\alpha=180$ deg



Écoulement autour d'un cylindre 2D

- Plusieurs modèles de turbulence disponibles dans FLUENT:

- RANS (modèles statistiques)
- LES (Large Eddy Simulation)

- Modèles statistiques :

- Spalart Almaras
- k-ε
- k-ω (Wilcox 1972)
- Rij (second ordre)

$\omega = \epsilon / (\beta * k)$ est la dissipation spécifique

$$\mu_t = \rho \frac{k}{\omega}$$

$$\rho \left\{ \frac{\partial k}{\partial t} + \bar{u}_j \frac{\partial k}{\partial x_j} \right\} = \frac{\partial}{\partial x_j} \left\{ (\mu + \mu_t \sigma_*) \frac{\partial k}{\partial x_j} \right\} + P - \rho \beta_* k \omega$$

$$\rho \left\{ \frac{\partial \omega}{\partial t} + \bar{u}_j \frac{\partial \omega}{\partial x_j} \right\} = \frac{\partial}{\partial x_j} \left\{ (\mu + \mu_t \sigma) \frac{\partial \omega}{\partial x_j} \right\} + \alpha \frac{\omega}{k} P - \rho \beta \omega^2$$

Constantes (universelles) du modèle k-ω

α	β	β_*	σ	σ_*
5/9	3/40	9/100	1/2	1/2

Suivi lagrangien de particules

- Les trajectoires de particules peuvent être suivies classiquement par :
 - Modèle eulérien : les particules sont considérées comme une phase supplémentaire continue – Ajout d'une (ou plusieurs) équation pour suivre l'évolution d'une nouvelle quantité qui définit la concentration massique locale en particules
 - Modèle lagrangien : on suit la trajectoire de chaque particule individuellement en résolvant le PFD appliqué à chacune d'entre elles.
- Modèle Lagrangien

$$\rho_p V_p \frac{d\vec{u}_p}{dt} = \frac{1}{2} \rho (U - u_p)^2 \frac{\vec{U} - \vec{u}_p}{|\vec{U} - \vec{u}_p|} C_D S_p + (\rho - \rho_p) V_p \vec{g} + \dots$$

Forces
supplémentaires

- La particule est supposée sphérique et la loi de traînée utilisée est généralement la loi de Stokes :

$$C_D = \frac{24}{Re_p} \quad \text{pour } Re_p < 1$$

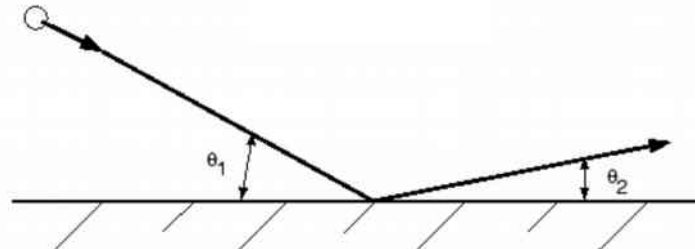
- Il faut définir la position et la vitesse initiale ! Concept d' **injection** dans fluide

Suivi lagrangien de particules

- Conditions aux limites :
 - Sur les parois :
 - Captation par la paroi (possibilité évaporation)
 - Réflexion

Coefficients de restitution (valeurs par défaut)

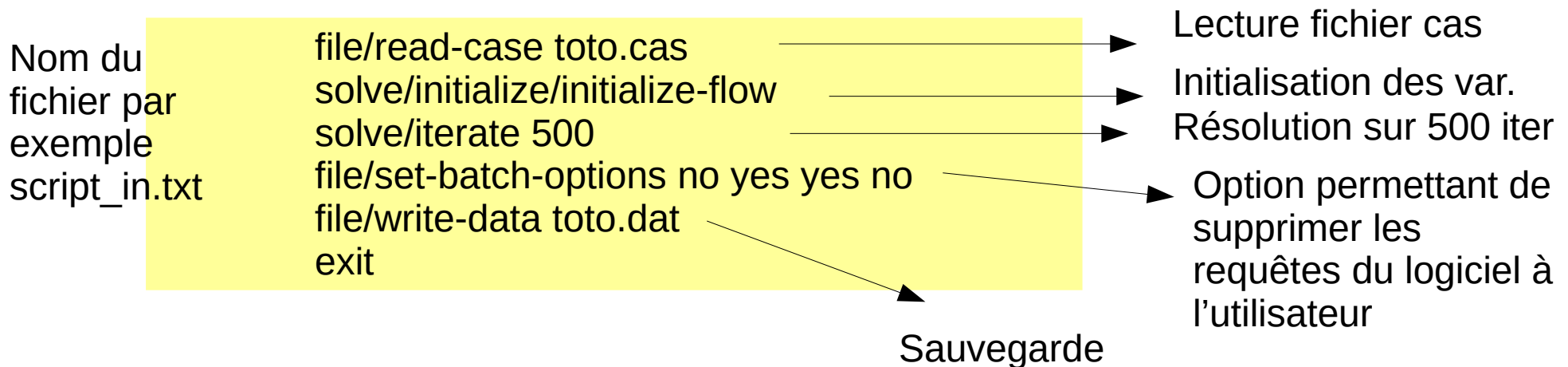
$$e_n = \frac{V_{2n}}{V_{1n}} = 1$$
$$e_t = \frac{V_{2t}}{V_{1t}} = 1$$



- Disparition (fin calcul trajectoire)
 - « Wall film »
 - « Wall jet »
 - « Interior » paroi transparente....
- Intégration numérique :
 - Analytique (efficace sauf lorsque le pas de temps est grand)
 - Schémas numériques implicites, trapézoïdaux, RK....
 - Dans les calculs d'écoulements instationnaires les pas de temps du pb fluide et de la phase discrète peuvent être différents.

Calcul en batch

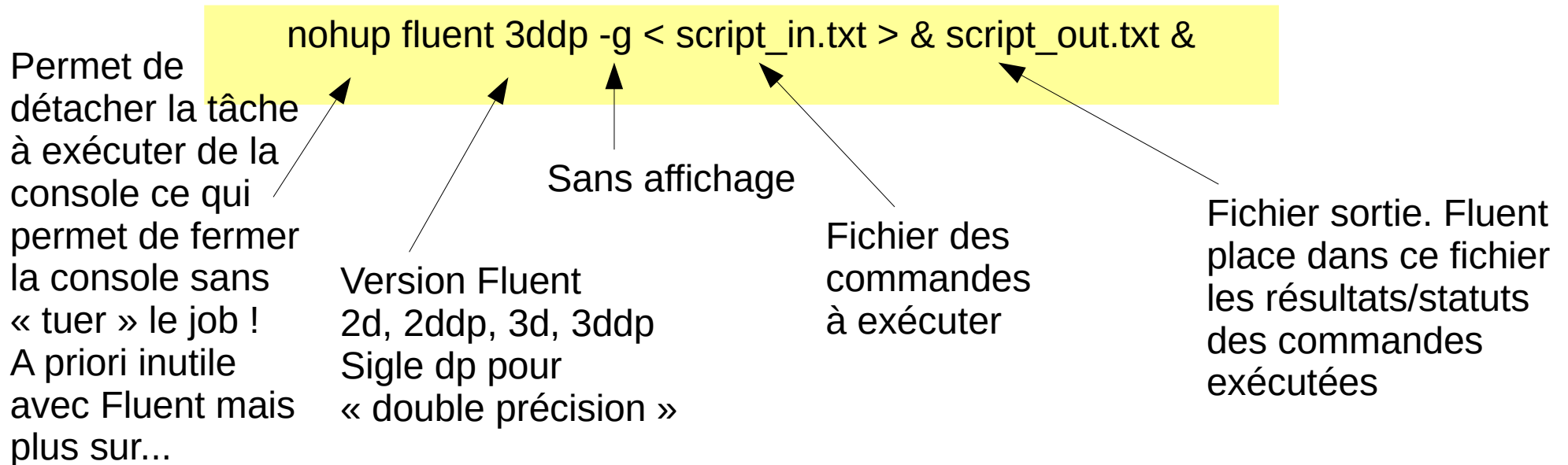
- Les calculs en batch sont utiles lorsque leur exécution dépasse plusieurs dizaines de minutes afin d'éviter de bloquer une console. On peut ainsi lancer le calcul puis se dé-logger.
- Pour exécuter le calcul en batch il faut préparer un fichier texte pour définir les commandes d'exécution de fluent



- Ce fichier peut être préparé dans un éditeur très simple comme gedit. (exécution en utilisant la commande gedit sur un terminal)
- Pour exécuter fluent en batch :

Calcul en batch en local

- Pour exécuter fluent en batch sur la machine locale (où vous êtes loggé):



- On peut par la suite se déconnecter et le « job » continuera d'être exécuter.
- Le fichier « script_out.txt » est continûment renseigné par fluent au cours de l'exécution. On peut se référer, en temps réel, aux dernières lignes de ce fichier pour connaître le statut du calcul :

```
tail -f script_out.txt
```

(CTRL + c pour quitter le programme tail)

Calcul en batch en remote

- Pour exécuter fluent en batch sur un serveur distant si vous êtes déjà dans le réseau de l'école :

```
ssh votre_identifiant@scylla
```

- Cette commande ouvre une session distante (remote) sur le serveur. On peut lancer le calcul avec la commande :

```
nohup fluent 3ddp -g < script_in.txt > & script_out.txt &
```

- On peut par la suite se déconnecter et le « job » continuera d'être exécuter.
- Si vous n'êtes pas dans le réseau de l'école (par exemple chez vous ou sur eduroam) il faut d'abord se connecter sur le « sas » qui permet d'accéder au réseau de l'école. Puis faire comme si vous étiez physiquement à l'école.

```
ssh votre\_identifiant@sas1.centrale-marseille.fr  
ssh votre_identifiant@scylla
```

Personnalisation du code

- Il est possible de « personnaliser » certaines fonctionnalités de Fluent en ajoutant des portions de code écrites en macro langage C :

Ce sont les udf ou User Define Function

- Les *udf* permettent de :
 - Réaliser un monitoring « à la carte » d'un calcul
 - Modifier des propriétés physiques
 - Définir des conditions aux limites variables en temps et en espace
 - Enrichir/modifier des modèles physiques
 - Réaliser des post-traitements particuliers
- Les udf sont écrites en C et compilées puis liées avec le code FLUENT
- Il est possible également d'utiliser l'interpréteur de commande mais le fonctionnement est globalement plus lent. Mais cela peut être utile pour debugger.

Personnalisation du code

- Exemple définir une condition aux limites « maison » !

```
#include "udf.h"
DEFINE_PROFILE(pressure_profile, t, i)
{
    real x[ND_ND];
    /* this will hold the position vector */
    real y;
    face_t f;
    begin_f_loop(f, t)
    {
        F_CENTROID(x,f,t);
        y = x[1];
        F_PROFILE(f, t, i) = 1.1e5 - y*y/(.0745*.0745)*0.1e5;
    }
    end_f_loop(f, t)
}
```

Librairie fluent

Macro permettant d'effectuer une boucle sur l'ensemble des éléments de la face frontière considérée

Nom de la fonction

Fonction pour récupérer les coord de la face courante

- Compilation avec Define/User-Defined/Functions/Compiled

Personnalisation du code

Udf pour définir la vitesse d'injection des aérosols

```
DEFINE_DPM_INJECTION_INIT(vit_aerosols,I)
{
  particle *p;
  real x ;

  real time = CURRENT_TIME;

  Message("Initializing Injection: %s\n",I->name);

  /* Injection faite uniquement pendant l'inspiration*/
  loop(p, I->p_init)
  {
    x=P_POS(p)[0] ;
    P_VEL(p)[2] = x;
    P_VEL(p)[1] = 0.0;
    P_VEL(p)[0] = 0.0;
  }
  Message("x=%f - U=%f \n",P_POS(p)[0],P_VEL(p)[2]);
}
```