HAL
open science

Simulation des grandes échelles (SGE) d'écoulements de fluides incompressibles Large-eddy simulation (LES) of incompressible fluid flows.

Boris Brangeon

To cite this version:

Boris Brangeon. Simulation des grandes échelles (SGE) d'écoulements de fluides incompressibles Large-eddy simulation (LES) of incompressible fluid flows.. 2008. hal-00682561

HAL Id: hal-00682561

https://hal.science/hal-00682561v1

Submitted on 26 Mar 2012

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

SIMULATION DES GRANDES ECHELLES D'ECOULEMENTS TURBULENTS AVEC Open∇FOAM

Comportement d'un écoulement turbulent autour d'un bâtiment

présenté et soutenu par
Boris BRANGEON
boris.brangeon@univ-pau.fr

Préparé au
Laboratoire de Physique du Bâtiment et des Systèmes (LPBS, Réunion)

en vue de l'obtention du
Master 2 Ingénieurie des Systèmes Thermiques (ISI, Pau)

Septembre , 2008

© Boris BRANGEON,2008.
Typset LaTeX

Remerciements

La qualité de ce stage n'aurait pas été aussi agréable sans la rencontre ni l'aide de nombreuses personnes. Aussi, je vais prendre le temps de citer toutes ces personnes, en priant d'avance celles et ceux que j'aurais oubliés, de bien vouloir m'en excuser.

Tout d'abord, je tiens à remercier M. François GARDE, directeur du laboratoire, pour m'avoir accueilli au sein de son équipe de chercheurs.

Je tiens ensuite à exprimer ma profonde reconnaissance à M. Alain BASTIDE, maître de conférence et maître de stage, pour son encadrement, pour le suivi qu'il a réalisé pendant ces six mois de ce travail de stage, pour sa disponibilité et ses précieux conseils. Il restera pour ma part un exemple de ce que doit être la pédagogie dans le secteur de l'enseignement. Il a largement contribué a me donner les moyens de réussir ce stage.

Je remercie également M. Jean-Philippe PRAENE et M. Dominique MORAU, post-doc, pour leurs conseils et leurs expériences professionnelles enrichissantes.

Enfin, je tiens à remercier l'ensemble du personnel du LPBS et particulièrement les thésards (Placide, Fakra), les stagiaires (Rémi, Thomas, Jean-Philippe, Neria-Isabelle, Volana...), la secrétaire de gestion (Vanessa),.... Ils ont rendu amicale l'ambiance de travail.

Nomenclature

Lettres latines

Symbole	*Signification*	*Unité SI*
D	Coefficient artificiel de diffusion	$-$
f	Force volumique	$N.m^{-3}$
k	Energie cinétique turbulente	$m^2.s^{-2}$
G_Δ	Opérateur de Filtrage	$-$
L	Echelle de longueur intégrale	m
p	Pression	Pa
Re	Nombre de Reynolds	$adim.$
S	Section entrée et sortie du bâtiment	m^2
S_{ij}	Tenseur de déformation	Pa
t	Temps	s
T	Echelle de temps	s
u	Vitesse	$m.s^{-1}$
y	Direction normale à la paroi	
V	Volume	m^3

Lettres grecques

Symbole	*Signification*	*Unité SI*
Δ	Taille du filtre caractéristique en LES	$adim$
Δx_i	Tailles de maille dans les differentes directions de l'espace	m
η	Echelle de longeur de Kolmokorov	m
ε	Dissipation de l'énergie turbulente	$m^2.s^{-3}$
$\vec{\Omega}$	Vecteur tourbillon	$-$
λ	Echelle de longueur de Taylor	m
λ_t	Echelle de temps de Taylor	s
ν	Viscosité cinématique	$m^2.s^{-1}$
ν_t	Viscosité cinématique turbulente	$m^2.s^{-1}$
μ	Viscosité dynamique	$kg.m^{-1}.s^{-1}$
ρ	Masse volumique	$kg.m^{-3}$
σ_{ij}	Tenseur des contraintes visqueuses	Pa
τ	Echelle de temps de Kolmokorov	s
τ_{ij}	Tenseur des contraintes de sous-maille	Pa

Constantes

Symbole	*Signification*	*Valeur*
C_k	Constante de Kolmokorov	$\sim 1,5$
A^+	Constante	26

Exposants

$\overline{()}$	quantité filtrée
$()^+$	grandeur rendue adimensionnelle en unité de paroi
$()'$	fluctuation
$()^T$	transposé

Abréviations

GNU	Gnu's Not Unix
LES	Large Eddy Simulation
RANS	Reynolds Averaged Navier-Stokes
VTK	Visualization ToolKit

Table des matières

Nomenclature iii

1 Introduction **1**

2 Modélisation des écoulements turbulents **4**
2.1 Démarche mathématique de la turbulence 4
2.1.1 Équation de la continuité 4
2.1.2 Équation de la dynamique 5
2.2 Démarche physique de la turbulence 6
2.3 Démarche numérique de la turbulence 6

3 La simulation numérique des grandes échelles de la turbulence, SGE **10**
3.1 Principes et avantages 10
3.2 Cascade d'énergie 10
3.3 Hypothèses de Kolmogorov 11
3.4 Notions de filtrages 12
3.5 Les équations de Navier-Stokes filtrées 13
3.6 Modélisation sous-maille 14
3.6.1 Viscosité turbulente 14
3.6.2 Modèle de Smagorinsky 15

4 Modélisation numérique utilisant la LES **17**
4.1 Introduction 17
4.2 Position du problème 17
4.3 Outil informatique : Open∇FOAM 19
4.3.1 Introduction 19
4.3.2 FoamX 20
4.4 Outil de visualisation : ParaView 21
4.5 Maillage 22
4.5.1 Introduction 22
4.5.2 Structure du fichier 22
4.5.3 Méthode de conception 23
4.5.4 Conclusion 28

4.6 Détermination des échelles de turbulence pour la simulation LES 28
4.6.1 Introduction 28
4.6.2 Modèle RNG-k-epsilon 28
4.6.3 Paramètres k et ε 29
4.6.4 Résultats de la simulation 30
4.6.5 Échelles de turbulence 30
4.7 Application de la LES à la convection turbulente sur un bâtiment ouvert en 2D . 34
4.7.1 Introduction 34
4.7.2 Algorithme de calcul 34
4.7.3 Discrétisations 36
4.7.4 Récapitulatif 38
4.8 Outils informatiques créés 38
4.8.1 Moyennes temporelles 38
4.8.2 Traitements des données 39
4.9 Résultats et discussions 42
4.9.1 Introduction 42
4.9.2 Champ de vitesse instantanée 42
4.9.3 Champ de vitesse moyennée 44
4.9.4 Etude de l'écoulement 44
4.9.5 Débit massique 46

Conclusion générale et perspectives **48**

A ANNEXES **50**

Bibliographie **53**

Table des figures

1.1 Photographie d'une tornade (photo Don Llyod 1996) 1
1.2 Vue satellite du cyclone Gamède (Réunion, 2007) 2

2.1 Exemple de mesure du champ de vitesse derrière un cylindre à Re=10000 [1] . 9

3.1 Mécanismes énergétiques 11
3.2 Simulation numérique de la LES 12

4.1 Domaine étudié 18
4.2 Aperçu de la structure d'OpenFOAM 19
4.3 Organigramme de fonctionnement 20
4.4 Interface graphique JAVA de FoamX 20
4.5 Illustrations de FoamX 21
4.6 Interface graphique de ParaView 22
4.7 Bâtiment élaboré dans GeoGebra 24
4.8 Schéma d'un bloc 25
4.9 Coefficient d'expansion (*source : User Guide OpenFOAM*) 25
4.10 Illustrations de la commande simpleGrading 26
4.11 Maillage du domaine d'étude 27
4.12 Paramètres k et ε 30
4.13 Echelles turbulentes 33
4.14 Champ instantanée de la vitesse 43
4.15 Champ moyenné de la vitesse 44
4.16 Type d'écoulement 45
4.17 Débit en fonction du temps 46
4.18 Evolution du débit 47

Liste des tableaux

4.1 Structure du maillage 23
4.2 Statistique du maillage 28
4.3 Échelles obtenues pour le maillage 33
4.4 Configuration du fichier de contrôle 38
4.5 Fichier de données généré par OpenFOAM 40
4.6 Mise en forme des données 40

Chapitre 1

Introduction

Les écoulements de fluides rencontrés dans la nature (l'eau qui s'écoule dans une rivière, atmosphère : cyclones, tornades, océans, fleuves, avalanches. . . . etc) (Figure 1.1 et Figure 1.2) ou dans les processus industriels (aéronautique, automobile, jet d'un réacteur, fumée qui sort d'une cheminée d'usine, etc . . .), sont en général instationnaires et turbulents, souvent couplés avec d'autres phénomènes physiques. La prise en considération de ces phénomènes instationnaires et de ces couplages est indispensable pour la compréhension et la prédiction de certains problèmes de la mécanique et de la physique. L'étude de la turbulence est un des problèmes majeurs de la recherche actuelle.

FIG. 1.1 – Photographie d'une tornade (photo Don Llyod 1996)

FIG. 1.2 – Vue satellite du cyclone Gamède (Réunion, 2007)

La turbulence n'est pas une propriété du fluide, comme la viscosité ou la conductivité thermique, par exemple. La turbulence désigne l'état d'un fluide, liquide ou gaz, dans lequel la vitesse présente en tout point un caractère tourbillonnaire : tourbillons dont la taille, la localisation et l'orientation varient constamment. Les écoulements turbulents se caractérisent donc par une apparence très désordonnée, un comportement difficilement prévisible et par la présence de nombreuses échelles spatiales et temporelles. De tels écoulements apparaissent lorsque la source d'énergie cinétique qui met le fluide en mouvement est relativement intense devant les forces de viscosité que le fluide oppose pour se déplacer. Pour traduire le fait que, dans un écoulement, les forces d'inertie l'emportent sur les forces de viscosité, le nombre de Reynolds doit être supérieur à un certain seuil (Re>2000).

Les équations qui régissent les écoulements turbulents sont les équations de Navier- Stokes (1822), qui se déduisent des lois newtoniennent de l'hydrodynamique. Le caractère non-linéaire du terme de convection qui apparaît dans ces équations est à l'origine des difficultés rencontrées dans la résolution de ce système. Du point de vue mathématique, on ne connaît des solutions analytiques que dans des cas particuliers (les écoulements de Couette ou de Poiseuille). Sinon, la théorie mathématique de ces équations est encore incomplète et fait partie de la catégorie des problèmes du prix du millénaire tels que les hypothèses de Riemann, conjecture de Poincaré, ...etc. Il n'existe que des résultats partiels d'existence, unicité, régularité de la solution, différents selon la dimension de l'espace considéré.

Ainsi, un calcul direct d'un écoulement turbulent permettrait d'avoir tous les détails des variables de l'écoulement mais il ne peut se concevoir pour la prévision des écoulements réels. Le traitement des configurations complexes ou à des grands nombres de Reynolds impliquerait un coût de calcul en taille mémoire et en en temps CPU trop important. On constate, de nos jours, que l'utilisation d'outils de simulation numérique pour l'étude d'écoulements dans le cadre d'applications industrielles n'a cessé d'augmenter. Cette progression a en partie été rendue possible grâce à la constante augmentation de la rapidité des processeurs, et de la réduction des coûts de composants électroniques à performances égales.

Lorsque l'on veut simuler des écoulements instationnaires (à grand nombre de Reynolds), la méthode de Simulation des Grandes Échelles (en anglais LES pour Large Eddy Simulation)

est bien adaptée. C'est la méthode que nous avons choisi pour résoudre les problèmes traités dans ce rapport. A l'aide de celle-ci, nous allons étudier le comportement d'un écoulement turbulent autour d'un bâtiment.

Ce stage de six mois, c'est effectué au sein du Laboratoire de Physique du Bâtiment et des Systèmes (LPBS) à la Réunion (974).

Les principaux objectifs de ce stage, qui ont permis l'orientation des travaux de recherche effectués dans ce rapport, sont les suivants :

- l'étude théorique et pratique de la LES sur des cas tests ;
- le développement d'outils informatiques, décrivant au mieux la physique de la turbulence ;
- l'étude des résultats obtenus par des simulations LES.

Pour atteindre ces objectifs, nous avons menés une analyse détaillée de la LES. Le plan proposé pour la présentation du travail est le suivant :

Dans le premier chapitre, nous présentons les démarches essentielles pour la compréhension de la turbulence. Le deuxième chapitre est consacré à la présentation de la méthode de simulation des grandes échelles. Dans le chapitre trois de ce rapport, on expose d'abord la problématique du sujet. Nous présentons ensuite la méthode qui a été suivit et nous évaluons les principaux résultats obtenus par nos modélisations. Enfin, des conclusions sur les résultats et les perspectives que nous envisageons dans ce domaine de recherche sont présentées.

Chapitre 2

Modélisation des écoulements turbulents

Dans cette étude, nous nous intéresserons donc ici essentiellement à la **mécanique des milieux continus** qui concerne la déformation des écoulements des fluides.
L'hypothèse des milieux continus consiste à considérer des milieux dont les propriétés caractéristiques sont continues. Une telle hypothèse permet d'avoir recours aux outils mathématiques reposant sur les fonctions continues et/ou dérivables [2].
Nous admettrons également les hypothèses suivantes :

- le fluide, ici de l'air, est un fluide **newtonien** qui est caractérisé par un coefficient de viscosité qui dépend de la température et de la pression.
- le fluide est **homogène**, ses propriétés sont les mêmes en tout point ;
- le fluide est **isotrope**, ses propriétés ne dépendent pas du repère dans lequel elles sont observées ou mesurées.

2.1 Démarche mathématique de la turbulence

Il est admis que le comportement de tout écoulement vérifiant l'hypothèse des milieux continus, quelle que soit la nature du fluide (compressible ou non), et de l'écoulement (turbulent, laminaire ou en transition), peut être représenté par les équations de Navier-Stokes (Annexe A) qui expriment la conservation de la quantité de mouvement, auxquelles viennent s'ajouter les équations de conservation de la masse et de l'énergie totale [3].
Dans cette étude, on suppose un fluide incompressible, à propriétés thermodynamiques constantes (μ = cte, η = cte, ρ = cte). On suppose qu'il n'y a pas de transfert de chaleur.

2.1.1 Équation de la continuité

L'équation de continuité s'écrit de la forme suivante :

$$\frac{D\rho}{Dt} + \rho\, div\vec{V} = 0 \tag{2.1}$$

Dans le cas des écoulements incompressibles à masse volumique constante, on a $\frac{D\rho}{Dt} = 0$, d'où $div\vec{V} = 0$ (on dit que le mouvement est isochore).

Si nous utilisons la convention d'Einstein sur les indices répétés, que adopterons par la suite, comme notation par défaut, on obtient alors :

$$\boxed{\frac{\partial u_i}{\partial x_i} = 0 \qquad , i = 1,2,3 \qquad (2.2)}$$

Ou encore sous la forme plus connue :

$$\nabla u = 0 \tag{2.3}$$

2.1.2 Équation de la dynamique

La forme générale de l'équation de Navier-Stokes traduisant la conservation de quantité de mouvement s'écrit (à masse volumique constante) :

$$\rho \frac{\partial u_i}{\partial t} + \rho u_j \frac{\partial u_i}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{\partial \sigma_{ij}}{\partial x_j} + \rho f_i \qquad , i = 1,2,3 \tag{2.4}$$

où f_i désigne une force.

Dans le cadre des fluides visqueux newtoniens, le tenseur des contraintes visqueuses s'écrit de la façon suivante [4] :

$$\sigma_{ij} = \mu \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3} \mu \frac{\partial u_j}{\partial x_j} \delta_{ij} \qquad , i = 1,2,3 \tag{2.5}$$

D'après l'équation (2.2), le second terme de l'équation (2.5) est nul.

On note le tenseur des taux de déformation :

$$S_{ij} = \frac{1}{2} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) \qquad , i = 1,2,3 \tag{2.6}$$

En faisant intervenir l'équation de continuité pour le calcul de σ_{ij} l'équation de Navier-Stokes incompressible s'écrit finalement sous la forme suivante :

$$\boxed{\frac{\partial u_i}{\partial t} + u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho} \frac{\partial p}{\partial x_i} + \frac{\partial}{\partial x_j}(2\nu S_{ij}) + f_i \qquad , i = 1,2,3 \qquad (2.7)}$$

Ou encore sous la forme plus connue :

$$(\partial_t + u.\nabla)u = -\frac{1}{\rho} \nabla p + \nu \Delta u + f \tag{2.8}$$

Les inconnues sont la vitesse $u = (u_1, u_2, u_3)$ et la pression p du fluide visqueux incompressible. Toutes les quantités sont évalués en (x,t), où x désigne la position et t représente le temps. f représente les forces volumiques appliquées au fluide et ν la viscosité cinématique. Le terme $(u.\nabla)u$ est le **terme de transport ou de convection** et le terme $\nu\Delta u$ est le **terme de diffusion** (interactions au niveau moléculaires entre les particules se traduisant par une dissipation d'énergie).

2.2 Démarche physique de la turbulence

Cette approche physique, nous permettra de mieux comprendre les phénomènes turbulents qui peuvent être caractérisés par un certain nombre de propriétés physiques, comme par exemple :

- **irrégularité en espace et en temps :** la vitesse et la pression varient de façon aléatoire. On ne peut pas décrire un mouvement turbulent dans tous ses détails comme une fonction du temps et de la position mais il est possible d'en faire une étude statistique en calculant les valeurs moyennes de vitesse et de pression,

- **structures :** dans un écoulement turbulent, on constate que des structures de tailles différentes interagissent entre elles,

- **diffusivité élevée :** est une propriété du régime de l'écoulement ; un champ turbulent diffuse fortement toute quantité transportable (température, quantité de mouvement),

- **écoulement rotationnel :** un écoulement turbulent présente toujours des fluctuations du rotationnel de la vitesse (voir phénomène de *vorticité*),

- **dissipation :** les taux de déformations étant très importants, ils contribuent à une forte conversion de l'énergie cinétique turbulente en chaleur,

- **caractère chaotique :** on ne peut pas prédire l'évolution ultérieure de l'écoulement pour des longs temps ; les mouvements turbulents sont non prédictibles.

2.3 Démarche numérique de la turbulence

La simulation numérique est devenue l'un des principaux outils de recherche de la mécanique des fluides et des transferts. Elle a permis, au fur et à mesure du développement des performances informatiques, de décrire des écoulements physiquement plus complexes. Ainsi, la prédétermination numérique des écoulements turbulents est très importante pour les applications industrielles.

La turbulence se caractérise par des fluctuations aléatoires des champs de vitesse et de pression, et ce sur un grand nombre d'échelles spatiales et temporelles différentes et inter-

dépendantes. Les limites de la simulation numérique se situent dans sa capacité à reproduire correctement des phénomènes aussi compliqués que la turbulence.

Les ordres de grandeur des échelles s'échelonnent depuis la taille caractéristique des plus grandes structures de l'écoulement considéré, L (échelle intégrale), jusqu'à celle des plus petites structures dynamiquement actives, correspondant à l'échelle dissipative de Kolmogorov η. A partir de la loi de Kolmokorov, on peut montrer que le rapport entre ces deux échelles est proportionnel à $Re^{3/4}$:

$$\frac{L}{\eta} = O(Re^{3/4}) \tag{2.9}$$

Ce qui implique que pour capturer toutes les échelles, il faut utiliser une grille de calcul avec un nombre de points dans chaque direction de l'espace proportionnel à $Re^{3/4}$, donc le nombre de points sur une grille tridimensionnelle sera de l'ordre de $Re^{9/4}$ et le temps de calcul est proportionnel à $Re^{11/4}$. Donc, calculer l'ensemble de ces échelles est extrêmement coûteux numériquement.

Afin de pouvoir résoudre les problèmes réels de turbulence, et donc les équations de la dynamique du fluide, il existe un déterminisme statistique. L'idée est de décomposer les grandeurs caractéristiques instantannées d'un écoulement turbulent en une partie **macroscopique** et une partie **turbulente aléatoire**. Cette décomposition permettra ensuite de développer un traitement statistique des équations de mouvement.

O. Reynolds a utilisé un opérateur de moyenne $\overline{()}$ (voir propriétés partie 3.4) et a proposé la décomposition des valeurs de la vitesse u et de la pression p en :

$$u(\vec{x},t) = \overline{u}(\vec{x},t) + u'(\vec{x},t) \qquad et \qquad p(\vec{x},t) = \overline{p}(\vec{x},t) + p'(\vec{x},t) \tag{2.10}$$

Dans l'analyse statistique des équations de Navier-Stokes, il existe trois types de moyenne utilisés [5] :

- **Moyenne d'ensemble :** c'est la moyenne des mesures de vitesse et de pression à chaque moment du temps et en chaque point de l'espace. on utilise ce type de traitement lorsque l'on fait de la simulation numérique directe (en anglais **D**irect **N**umerical **S**imulation ou **DNS**). La DNS semble être une première tentative de calcul de la turbulence mais elle ne peut pas se concevoir dans le but de prévisions numériques (météorologie) car elle demande des ressources informatiques importantes,

- **Moyenne temporelle :** est définie sur un échantillon et non pas sur un ensemble comme le cas précédent. On a alors :

$$\overline{u}(\vec{x},t) = \frac{1}{T}\int_{s=t}^{s=t+T} u(\vec{x},s)ds$$
$$\overline{p}(\vec{x},t) = \frac{1}{T}\int_{s=t}^{s=t+T} p(\vec{x},s)ds$$

Suite à la décomposition des équations de Navier-Stokes, il en résulte un tenseur de contraintes, appelé tenseur de Reynolds qui représente l'influence énergétique des petites échelles sur les valeurs moyennes de l'écoulement.
Enfin, après le traitement statistique des équations de Navier-Stokes, on a une perte d'informations sur les équations. L'information perdue doit être reconstruite, puis réintroduite dans le système à l'aide de méthode de fermeture qui ont pour but d'assurer la conservation des caractères fondamentaux (dissipation, diffusion, ...). Ces méthodes de fermeture en un point sont désignés en anglais sous le sigle **RANS** (**R**eynolds **A**vered **N**avier-**S**tokes),

- **Moyenne spatiale :** de façon similaire que le cas précédent , on obtient pour un écoulement homogène :

$$\overline{u}(\vec{x},t) = \frac{1}{|\Omega|}\int_{s=\Omega} u(s,t)ds$$
$$\overline{p}(\vec{x},t) = \frac{1}{|\Omega|}\int_{s=\Omega} p(s,t)ds$$

dans $\Omega \times [0,\infty[$.

Pour un écoulement où les champs sont inhomogènes, on utilise une moyenne spatiale filtrée (voir partie 3.4) :

$$\overline{u}(x,t) = \int_{\Omega} G_{\Delta}(x-x')\overline{u}(x',t)dx'$$

Après le traitement statistique, on obtient un système similaire que le cas précédent mais le tenseur de Reynolds n'a pas le même sens physique. Ici, il ne rend compte que des fluctuations des échelles plus petites que les mailles de calcul : on l'appelle aussi le <u>tenseur de sous-maille</u>. Les modèles de fermeture utilisés sont appelés <u>modèles de sous-maille</u>.

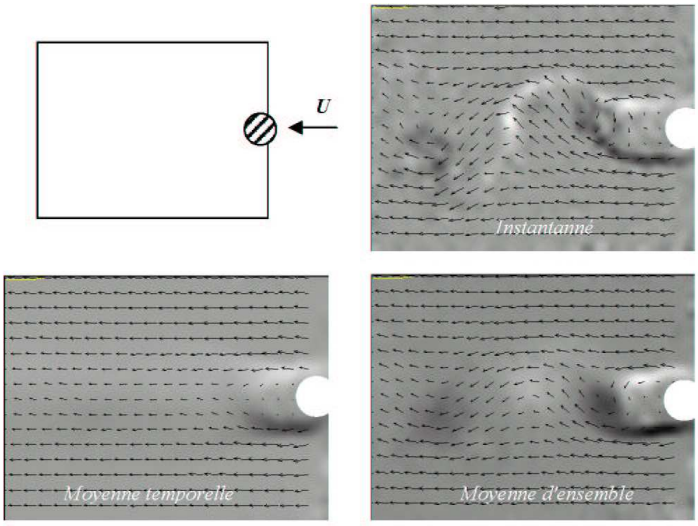


FIG. 2.1 – Exemple de mesure du champ de vitesse derrière un cylindre à Re=10000 [1]

Pour la modélisation de notre bâtiment, nous avons choisi de pratiquer une moyenne spatiale filtrée sur le champ turbulent. Cette résolution numérique porte le nom de **S**imulation des **G**randes **E**chelles (**SGE**), ou en anglais **L**arge **E**ddy **S**imulation (**LES**).

Chapitre 3

La simulation numérique des grandes échelles de la turbulence, SGE

3.1 Principes et avantages

Le principe de la simulation des grandes échelles est de calculer par **résolution numérique directe** les structures turbulentes de grande taille et de **modéliser** celles de petite taille. En effet, les grandes tourbillons sont très différents selon les géométries, leurs structures varient fortement selon l'écoulement considéré, par contre les petits tourbillons ont un caractère beaucoup plus universel.

Les avantages de cette méthodes sont les suivant :
- réduction de la puissance de calcul requise pour la simulation numérique direct,
- l'utilisation des modèles de fermeture donnent de bon résultat,
- utilisation d'écoulement plus complexe.

3.2 Cascade d'énergie

Une manière de schématiser simplement ce phénomène est l'analyse spectrale des équations de Navier-Stokes et sans entrer dans les détails, on rappelle juste l'expression du spectre de l'énergie cinétique dans l'espace de Fourier :

$$E(k) = \frac{1}{2}\sum |\hat{u}(k)|^2, \quad k \in \mathbb{N} \tag{3.1}$$

où : $\hat{u}(k)$ sont les coefficients de Fourier du champ de vitesse u.

et l'équation d'évolution de l'énergie cinétique dans l'espace spectral, obtenue à partir des équations de Navier-Stokes :

$$(\partial_t + 2\nu k^2)E(k) = W(k,t) + Z(k,t) \tag{3.2}$$

où :

- $W(k,t)$ est le terme qui représente la **production d'énergie** due aux forces appliquées au fluide,

- $Z(k,t)$ représente le **transfert d'énergie** dû aux termes non-linéaires des équations de Navier-Stokes.

A partir de l'équation (3.2), le concept de cascade d'énergie peut s'interpréter de la manière suivante : l'énergie introduite dans le système au niveau des grandes échelles par le terme $W(k,t)$, et est ensuite transférée aux échelles de plus en plus petites par le terme $Z(k,t)$, pour être enfin dissipée au niveau des plus petites échelles par le terme $2\nu k^2 E(k)$.
En effet, on considère que les structures turbulentes sont des éléments tourbillonnaires qui s'étirent les uns aux autres, produisant le transfert d'énergie à des échelles de plus en plus petites, jusqu'à ce que les forces visqueuses deviennent actives et dissipent l'énergie : ce processus, illustré par la figure 3.1, est connu sous le nom de **cascade d'énergie** [5].

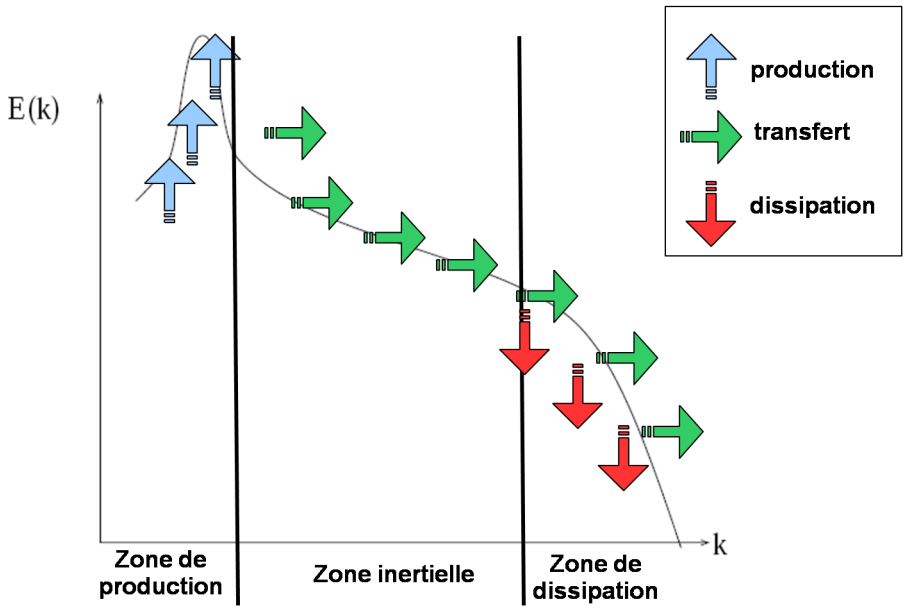


FIG. 3.1 – Mécanismes énergétiques

3.3 Hypothèses de Kolmogorov

Le modèle de spectre $E(k)$ de Kolmogorov (1941), est définit par deux hypothèses :

- L'énergie turbulente k est principalement portée par les grands tourbillons qui ne répondent pas directement aux effets de la viscosité,

- La viscosité moléculaire n'influence que sur les plus petites structures qui sont responsables de la majeure partie de la dissipation.

La théorie de Kolmogorov prédit que le spectre d'énergie cinétique turbulente peut s'écrire sous la forme : $E(k) \propto k^a \varepsilon^b$ et que la pente du spectre d'énergie dans la zone inertielle est égale à $b = -5/3$ en échelle logarithmique. Plus précisément, on a :

$$E(k) = C_k \varepsilon^{2/3} k^{-5/3} \tag{3.3}$$

où $C_k = 1,5$ est la constante de Kolmogorov et ε représente le taux de dissipation.

Dans cette partie, nous avons présenté l'analyse spectrale du système d'équation de Navier-Stokes qui nous permet d'introduire l'idée de petites échelles et de grandes échelles. En effet, la LES consiste à ne simuler que les échelles inférieures à *l'échelle dite de coupure* k_c, plutôt que de simuler numériquement toutes les échelles (DNS). Cette méthode de simulation est présentée par la figure suivante :

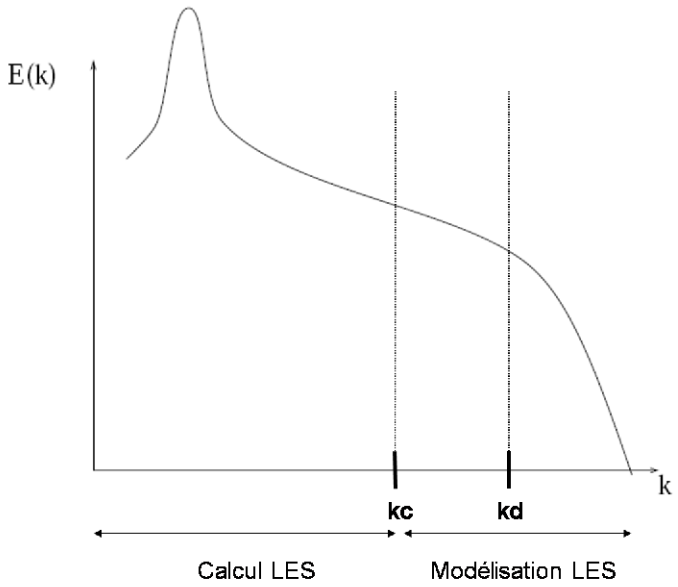


FIG. 3.2 – Simulation numérique de la LES

Dans la partie suivante, nous allons introduire la notion de filtrage, qui va nous permettre de distinguer ce qu'on entend par ***grandes échelles*** et ***petites échelles*** d'un écoulement turbulent.

3.4 Notions de filtrages

Le concept de modélisation de sous-maille repose sur une distinction entre les grandes et petites structures turbulentes, les premières devant être résolues alors que les secondes font l'objet d'une modélisation.

Cette séparation d'échelles est faite à l'aide d'une opération de filtrage passe-bas appliquée au système des équations régissant l'écoulement.

Si on note ψ une variable à filtrée, et G une fonction filtre de taille caractéristique Δ, alors la fonction filtrée $\overline{\psi}$, correspondant aux échelles de taille supérieures à Δ, est le résultat de l'opération suivante :

$$\overline{\psi}(x,t) = \int_{\Omega} G_{\Delta}(x,x')\overline{\psi}(x',t)dx' \tag{3.4}$$

où $\overline{\psi}$ correspond au champ résolu, $G_{\Delta}(x,x')$ désigne le filtre appliqué à la variable instantanée au point x et détermine la partie des fluctuations qui est incorporée dans le gros tourbillons. Δ est un paramètre qui représente la largeur du filtre choisi.

Ainsi, la partie haute fréquence de ψ non résolue, notée ψ', représente le champ de sous maille (ou partie fluctuante) et est définie par :

$$\psi' = \psi - \overline{\psi} \tag{3.5}$$

L'opérateur de filtrage est supposé normé :

$$\iiint_{\mathbb{R}^3} G_{\Delta}(x,x')dx' = 1 \tag{3.6}$$

En turbulence homogène, la grandeur filtrée ne dépend pas de la position x et donc, on peut écrire :

$$\overline{\psi} = \iiint_{\mathbb{R}^3} G_{\Delta}(x-x')\overline{\psi}(x',t)dx' \tag{3.7}$$

Cette équation représente en fait un produit de convolution $\overline{\psi} = G * \psi$.

La taille caractéristique du filtre Δ est définie par Germano et al [6] comme :

$$\Delta = (\Delta_x \Delta_y \Delta_z)^{1/3} \tag{3.8}$$

où $\Delta_x, \Delta_y, \Delta_z$ représentent respectivement les pas du maillage dans les trois directions de l'espace.

3.5 Les équations de Navier-Stokes filtrées

En appliquant un filtrage passe bas aux équations de Navier-Stokes (2.7 et 2.2) et avec les hypothèses précédentes, on obtient les équations régissant les variables moyennes $\overline{u}$ et $\overline{p}$, qu'on appelle les **équations de Navier-Stokes filtrées** :

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0 \qquad , i = 1,2,3 \tag{3.9}$$

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\overline{u}_j\overline{u}_i) = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}(2\nu\overline{S}_{ij} + \overline{\tau}_{ij}) + \overline{f}_i \qquad ,i = 1,2,3 \tag{3.10}$$

Un nouveau terme apparaît : $\overline{\tau}_{ij}$. Il est issu du filtrage et représente le tenseur des contraintes de sous-maille (ou tenseur de Reynolds). Il est défini par :

$$\overline{\tau}_{ij} = \overline{u}_i\overline{u}_j - \overline{u_i u_j} \tag{3.11}$$

Après avoir obtenu les équations de Navier-Stokes filtrées, on réalise la "*fermeture*" du système obtenu. Le problème de fermeture consiste à modéliser le tenseur des contraintes de sous-maille $\overline{\tau}_{ij}$, par des modèles qu'on appelle des **modèles de fermeture de sous-maille**.

3.6 Modélisation sous-maille

Dans cette partie, on cherche à modéliser l'influence des échelles plus petites que la maille de calcul, sur la dynamique des échelles résolues (échelles de grandes tailles).

3.6.1 Viscosité turbulente

La modélisation se base sur une hypothèse d'une dépendance linéaire entre les contraintes turbulentes de sous-maille, $\overline{\tau}_{ij}$ au tenseur des taux de déformation résolus, $\overline{S}_{ij}$, appelé **hypothèse de Boussinesq**.

$$\overline{\tau}_{ij} - \frac{1}{3}\overline{\tau}_{kk}\delta_{kk} = 2\nu_t\overline{S}_{ij} \tag{3.12}$$

avec :

- $\overline{S}_{ij} = \frac{1}{2}\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right)$ est le tenseur de déformation du champ filtré,
- ν_t est un coefficient appelé ***viscosité turbulente***.

La viscosité turbulente n'est pas une propriété du fluide mais une propriété du mouvement. Elle est calculé en chaque point du domaine de calcul.

A partir de l'équation (3.12), l'équation (3.10) peut s'écrire sous la forme :

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\overline{u}_j\overline{u}_i) = -\frac{1}{\rho}\frac{\partial \overline{P}}{\partial x_i} + 2\frac{\partial}{\partial x_j}\left[(\nu + \nu_t)\overline{S}_{ij}\right] + \overline{f}_i \qquad ,i = 1,2,3 \tag{3.13}$$

et

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0 \qquad ,i = 1,2,3 \tag{3.14}$$

où $\overline{P}$ est la pression modifiée $\overline{P} = \overline{p} - \frac{1}{3}\overline{\tau}_{kk}$.

Les modèles de viscosité turbulente consiste à modifier la viscosité de l'écoulement de façon qu'on puisse compenser l'absence d'une partie des termes non-linéaires d'interaction liés au petites échelles [5]. Il existe plusieurs modèles : modèles de viscosité spectrale, modèle de fonction de structure, modèles de sous-maille anisotropes, etc ..., mais dans la partie qui suit, on expose le modèle qui a été utilisé dans la modélisation LES du bâtiment (4.2), le modèle de Smagorinsky.

3.6.2 Modèle de Smagorinsky

C'est historiquement le premier modèle de de sous maille. Il est l'un des modèles les plus utilisés en LES. Il a été proposé par le météorologue américain J.Smagorinsky en 1963.

On fait l'hypothèse que la turbulence à petite échelle est isotrope, c'est à dire que nous avons une cascade de Kolmogorov en $k^{-5/3}$.

Ainsi, le modèle définit la viscosité turbulente de sous maille de la façon suivante :

$$\nu_t = (C_s\Delta)^2\|\overline{S}\| \tag{3.15}$$

où :

- $\|\overline{S}\|$ est la norme du tenseur des taux de déformation qui est défini par :

$$\|\overline{S}\| = \sqrt{2\overline{S}_{ij}\overline{S}_{ij}} \tag{3.16}$$

- C_s est la constante de Smagorinsky[1]. La valeur théorique de la constante du modèle de Smagorinsky peut être calculée dans le cadre d'une turbulence suivant un spectre de Kolmogorov, ayant alors pour valeur : $C_s = 0,18$. Bien qu'une valeur théorique puisse être donnée pour cette constante, dans la pratique cette valeur est modifiée afin de d'obtenir de meilleurs résultats : par exemple, Deardorff préconise par exemple la valeur de $C_s = 0,1$ (pour des écoulements cisaillés libres).

Le modèle de Smagorinsky possède quelques défauts, notamment celui de tendre vers une valeur non nulle lorsque que s'approche des parois. Pour contourner ce défaut, on utilise des fonctions d'amortissements permettant de ramener la valeur de ν_t à la paroi à 0. Par exemple, pour notre modélisation, nous avons utilisé la fonction de Van Driest qui se définit de la manière suivante :

$$f_{Van\,Driest} = 1 - \exp(-\frac{y^+}{A^+}) \tag{3.17}$$

avec :

[1] qui n'est pas une vraie constante mais une fonction de filtrage

- $y^+ = \frac{yu_{parietale}}{\nu}$ est la distance à la paroi,
- $A^+ = 26$, une constante.

L'équation (3.15) s'écrit de la façon suivante :

$$\nu_t = (C_s \Delta * f_{Van\,Driest})^2 \|\overline{S}\| \tag{3.18}$$

Chapitre 4

Modélisation numérique utilisant la LES

4.1 Introduction

Dans le chapitre précédent, nous avons introduit et expliqué la simulation des grandes échelles qui se base donc sur un découpage des fluctuations turbulentes. Une telle simulation permet de ne calculer que les grandes structures énergétiques et de modéliser les petites structures qui ont un caractères beaucoup plus universels.

Ce chapitre est consacré, dans un premier temps, à la présentation de notre démarche de travail, qui consiste à bien configurer une une simulation des grandes échelles. Dans un deuxième temps, nous mettrons en évidence certains phénomènes physiques rencontrés lors d'écoulement turbulent. D'ailleurs, pour illustrer ces phénomènes, nous nous appuierons sur un exemple d'écoulement turbulent autour d'un bâtiment. Nous proposons de suivre le plan suivant :

1. position du problème ;
2. présentation des outils informatiques OpenFOAM et ParaView ;
3. conception du maillage ;
4. détermination des échelles turbulentes ;
5. simulation LES d'un écoulement turbulent autour d'un bâtiment ;
6. résultats et discussions.

4.2 Position du problème

Dans le secteur du bâtiment, et plus particulièrement dans le domaine de la ventilation, l'utilisation de la LES n'est pas très répandue. Depuis quelques années, les performances informatiques ne cessent d'augmenter et permettent l'utilisation des simulations des grandes échelles (LES), souvent utilisées dans le secteur aéronautique. C'est pourquoi, il est intéressant d'étudier l'écoulement dynamique autour d'un bâtiment, avec cette approche.

Le bâtiment en question, peut être considéré comme un cube de côté H=3m ouvert sur deux côtés, comme le montre la figure 4.1. Le fluide newtonnien utilisé, dans l'écoulement, est de l'air, supposé incompressible. Nous ferons une étude d'un écoulement en 2D et nous utiliserons le logiciel OpenFOAM pour réaliser notre simulation.

Domaine

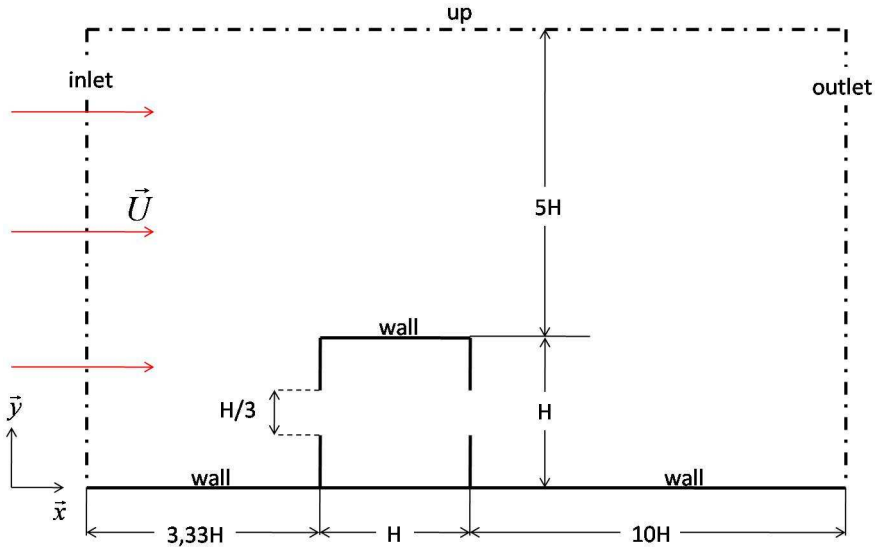


FIG. 4.1 – Domaine étudié

Formulation du problème

- étude en régime stationnaire et instationnaire,
- écoulement isotrope et homogène,
- fluide newtonien : air,
- Continuité de la masse :

$$\nabla U = 0 \tag{4.1}$$

- Conservation de quantité de mouvement :

$$(\partial_t + U.\nabla)U = -\frac{1}{\rho}\nabla p + \nabla.\left[(\nu + \nu_t)\nabla U\right] \tag{4.2}$$

Conditions limites

- `inlet` (à gauche sur le schéma) : U = (1,0,0) m.s^{-1},
- `outlet` (à droite sur le schéma) : p = 0 Pa,
- `wall` (paroi, sol) : U =(0,0,0),
- pour ramener le problème en 2D, car OpenFOAM ne traite que problèmes à volume finis, on utilise la condition limite `symmetryPlane` sur les autres surfaces.

Conditions initiales : U = 1 m.s^{-1}, p = 0 Pa,

Propriétés thermophysiques : $\nu = \mu/\rho = 18{,}1 \times 10^{-6}/1{,}293 = 14\ \mu m^2.s^{-1}$.

4.3 Outil informatique : Open∇FOAM

4.3.1 Introduction

Le logiciel de CFD utilisé, pour la réalisation de la simulation numérique en mécanique des fluides (**C**omputational **F**luid **D**ynamics, CFD), se nomme OpenFOAM. La boîte à outils d'OpenFOAM (**Open** **F**ield **O**peration **A**nd **M**anipulation), peut simuler des flux de fluide complexes comportant un transfert de réactions chimiques, de la turbulence et du transfert de chaleur, de l'électromagnétisme et de l'évaluation des options financières. OpenFOAM est produit par OpenCFD Ltd et est une source libre grand public de GNU. La technologie de base d'OpenFOAM est un ensemble flexible de modules efficaces de C++.

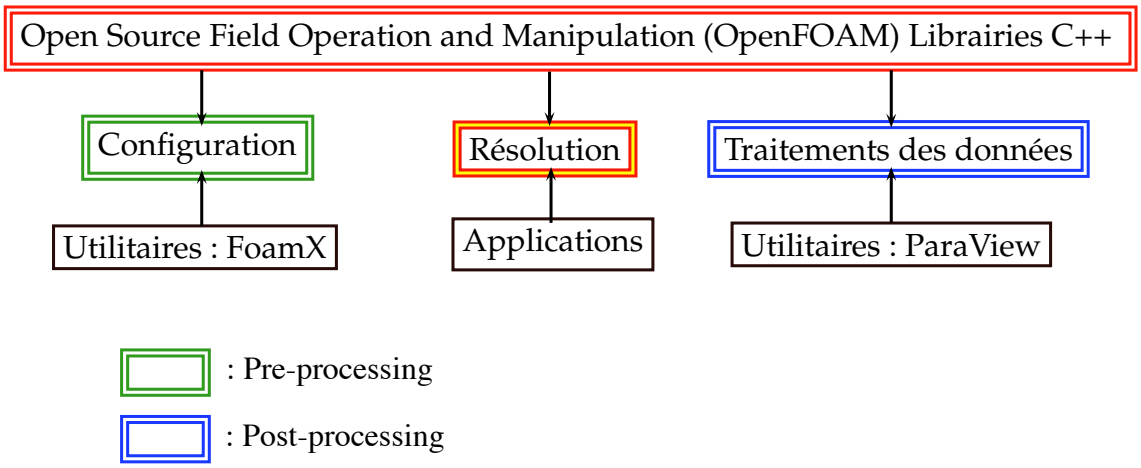


FIG. 4.2 – Aperçu de la structure d'OpenFOAM

OpenFOAM fonctionne à partir d'un langage, que l'on appelle de haut niveau, basé sur le C++. Il utilise des pointeurs d'adresses très performants. Ce langage m'étant inconnu, j'ai donc procédé à une exploration des librairies C++ (solveurs, utilitaires, options, etc ...) afin de comprendre le code et la structure du logiciel.

Le schéma suivant présente le fonctionnement d'OpenFOAM :

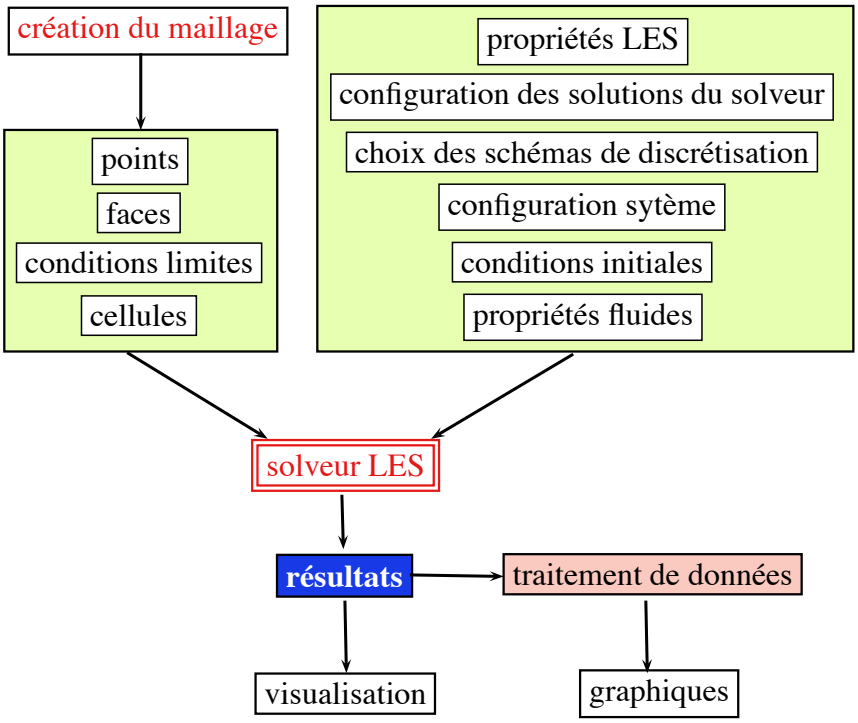


FIG. 4.3 – Organigramme de fonctionnement

4.3.2 FoamX

L'outil informatique FoamX, codé en langage JAVA, est une interface graphique d'OpenFOAM qui propose tous les utilitaires de "pre- et de post-processsing". C'est un outil simple destiné à un public non expérimenté.

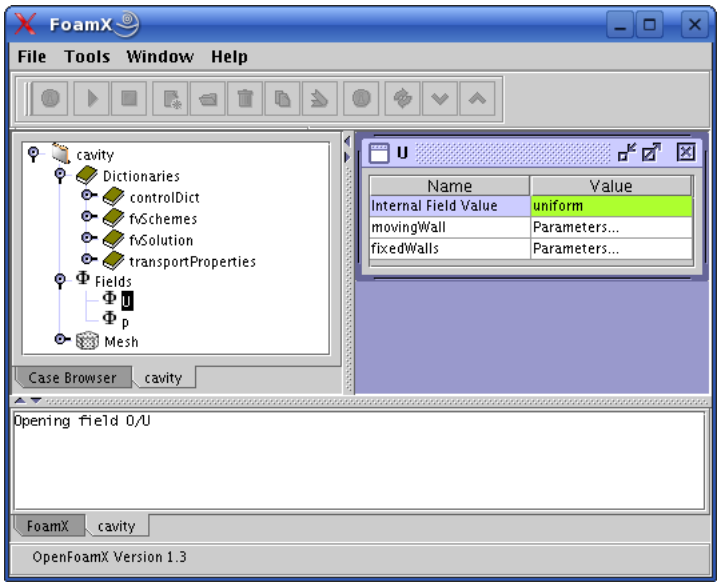
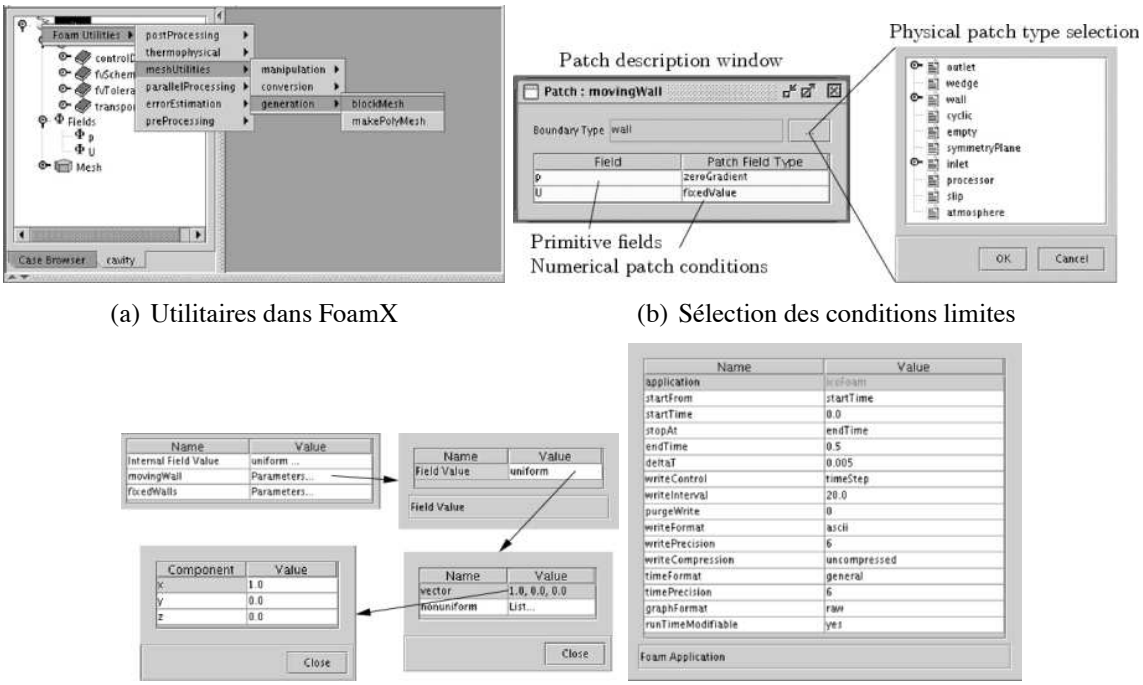


FIG. 4.4 – Interface graphique JAVA de FoamX



(a) Utilitaires dans FoamX

(b) Sélection des conditions limites

(c) Édition des conditions limites

(d) Exemple : fenêtre "ControlDict"

FIG. 4.5 – Illustrations de FoamX

Après exploration en détail d'OpenFOAM et une compréhension rapide du langage shell, l'outil FoamX, ne m'était plus indispensable.
En ce qui concerne la visualisation de mes résultats, j'ai utilisé le logiciel ParaView.

4.4 Outil de visualisation : ParaView

ParaView (Parallel Visualization Application) est un logiciel multi-plateformes basé sur VTK permettant la visualisation interactive de gros volumes de données scientifiques 2D et 3D. L'utilisateur construit intéractivement le pipeline graphique (succession de filtres agissant sur les données) de son choix, par exemple : qualité du maillage, champ de vecteurs, etc...
L'interface est ergonomique et elle est facile a prendre en main. Il suffit d'apprendre progressivement à connaître les actions des filtres sur les données (possibilité d'en ajouter de nouveaux). Ce logiciel permet de visualiser des données provenant de domaines très variés : écoulements, images médicales, molécules,....
La commande d'OpenFOAM, qui permet de visualiser les données sous ParaView est la suivante :

```
paraFoam -case (DossierSimu)
```

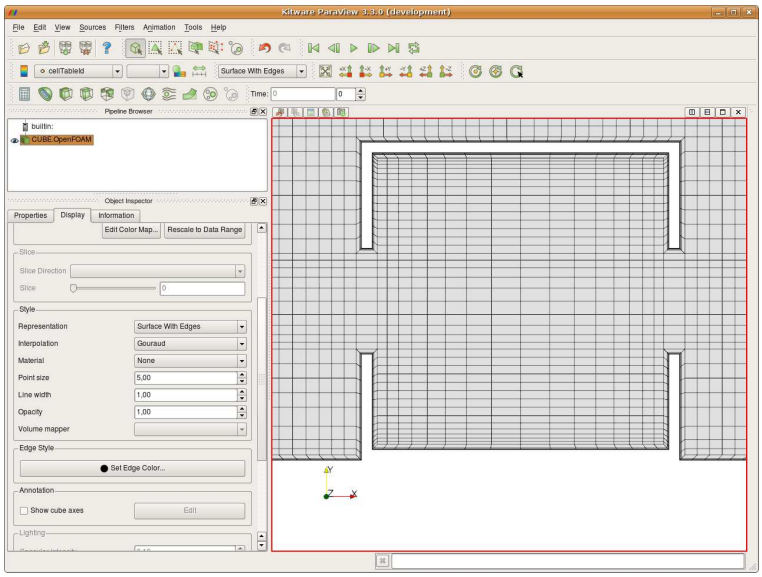


FIG. 4.6 – Interface graphique de ParaView

4.5 Maillage

4.5.1 Introduction

Dans OpenFOAM, il n'existe pas d'interface graphique pour confectionner les maillages. En revanche, le logiciel présente des outils qui sont capables de convertir, au format OpenFOAM, un maillage réalisé sur un autre logiciel : par exemple Fluent, Gambit, Star-CD,.... L'utilisation de la majeur partie des logiciels, cités précédemment, nécessite une licence payante. Par défaut de celle-ci, j'ai donc réalisé le maillage 2D du domaine étudié (voir figure 4.1) dans OpenFOAM. La partie suivante présente la méthode d'élaboration du maillage.

4.5.2 Structure du fichier

La structure du fichier (nommé `blockMeshDict.C`), qui génère un maillage dans OpenFOAM, est la suivante :

Élément	Mot clé OpenFOAM	Description	Exemple
unité	*convertToMeters*	spécifie l'échelle utilisée	0.001 pour des mm
points	*vertices*	listes des coordonnées des points	(0 0 0)
arêtes	*edges*	spécifie si l'arête est un arc, polyligne	arc 1 4 (1.1 0.0 0.5)
blocs de cellules	*block*	permet de créer et de spécifier le type de blocs de cellules (hexaèdre, prisme,...)	hex (0 1 2 3 4 5 6)
faces limites	*patches*	permet de créer et de spécifier le type de face (murs, entrée, sortie,...)	wall mur1 ((0 1 2 3))

TAB. 4.1 – Structure du maillage

4.5.3 Méthode de conception

Dans cette partie, j'explique comment j'ai procédé pour créer mon maillage 2D dans OpenFOAM. Dans un premier temps, nous expliquerons la méthode adoptée pour déterminer les coordonnées de chaque point du maillage. Puis, dans un second temps, nous expliquerons les quelques règles à respecter pour la réalisation d'un maillage.

En présence d'un domaine à géométrie complexe, il est difficile de déterminer les coordonnées des points sans passé par un logiciel de géométrie. Pour ma part, j'ai utilisé le freeware GeoGebra qui est un logiciel dynamique de mathématiques réunissant géométrie, algèbre et calcul différentiel. Il a été développé dans un but éducatif pour le secondaire par Markus Hohenwarter à l'Université de Salzburg (voir figure 4.7).

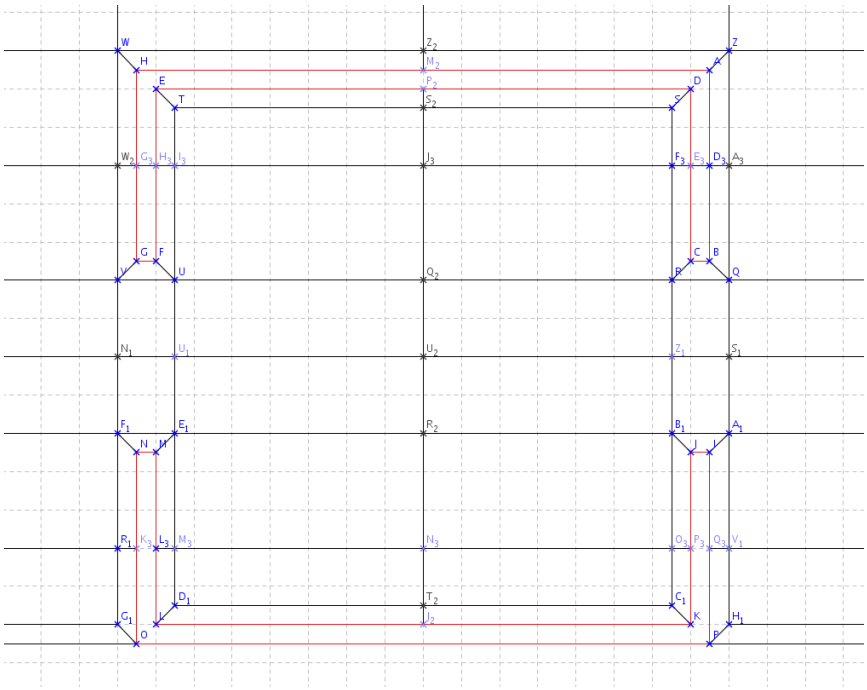


FIG. 4.7 – Bâtiment élaboré dans GeoGebra

Le maillage est exporté au format `.tex` , avec une mise en forme Pstricks (package LaTeX), et ensuite traité par l'intermédiaire d'un script bash (non explicité dans ce rapport), qui me récupère toutes les coordonnées du dessin et exportent celles-ci dans le fichier `blockMeshDict.C` avec la syntaxe OpenFOAM :

```
vertices
(
(1.5 1.5 -0.5) //A-->pt n° 0
(1.5 0.5 -0.5) //B
(1.4 0.5 -0.5) //C
(1.4 1.4 -0.5) //D
(-1.4 1.4 -0.5) //E-->pt n° 4
(. . . etc
);
```

Par la suite, il faut créer manuellement les blocs contenant les cellules (`blocks`) en respectant, pour la construction, le sens trigonométrique. Dans un repère direct, le sens trigonométrique va de l'axe (positif) de X vers Y (pour le plan xOy), de Y vers Z (pour yOz) et de Z vers X (zOx). Chaque bloc est définie par le libellé de ses sommets, exemple : le point A est définie par le libellé 0. La figure 4.8 et l'exemple ci-après, illustrent la construction d'un bloc :

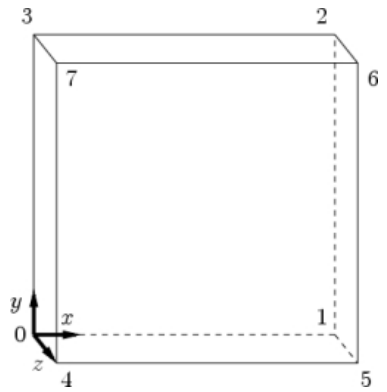


FIG. 4.8 – Schéma d'un bloc

Syntaxe dans OpenFOAM :

```
blocks
(
hex (0 1 2 3 4 5 6 7) (1 1 1) simpleGrading (1 1 1)
);
```

avec :

- `hex` définie le type de bloc, ici il s'agit d'un hexaèdre,
- (0 1 2 3 4 5 6 7) désigne l'ordre de construction du polyèdre à six faces, par exemple la combinaison (0 3 2 1 4 7 6 5) n'est pas correcte car elle ne respecte pas le sens direct,
- (1 1 1) spécifie le nombre de cellules sur chaque direction $\vec{x}$,$\vec{y}$ et $\vec{z}$.

D'autre part, l'option `simpleGrading` permet d'appliquer un coefficient d'expansion égale à $\frac{\delta_e}{\delta_s}$. Il permet de constituer un maillage "adaptatif", c'est à dire, établir une maillage plus fin proche des parois du bâtiment vers un maillage plus grossier quand on s'en éloigne :

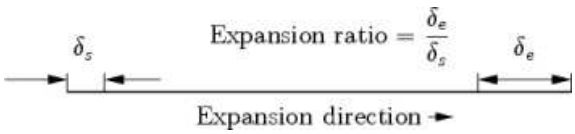
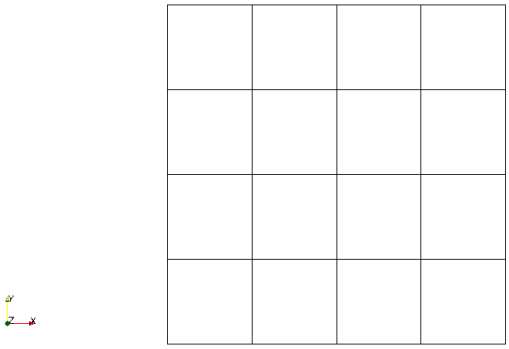
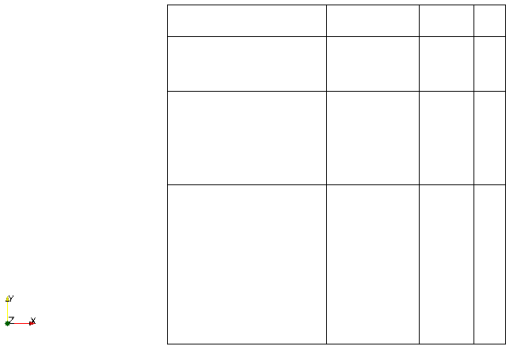


FIG. 4.9 – Coefficient d'expansion (*source : User Guide OpenFOAM*)



(a) hex (0 1 2 3 4 5 6 7) (4 4 1) simpleGrading (1 1 1)



(b) hex (0 1 2 3 4 5 6 7) (4 4 1) simpleGrading (0.2 0.2 1)

FIG. 4.10 – Illustrations de la commande simpleGrading

Enfin, il faut définir les faces (`patches`) des blocs confrontées à une conditions limites et leurs types : mur, entrée, sortie, ... Reprenons l'exemple de la figure 4.8.

```
patches                     // mot clé
    (
        patch                   // type de face
        inlet                   // nom du patch
        (
            (0 4 7 3)
        )

        patch
        outlet
        (
            (1 2 6 5)
        )

        wall
```

```
        walls
        (
            (0 1 5 4)
            (0 3 2 1)
            (3 7 6 2)
            (4 5 6 7)
        )
);
```

A partir du moment où l'on respecte la structure et les règles du fichier `blockMeshDict`, la commande d'OpenFOAM, qui permet de générer le maillage est la suivante :

```
blockMesh -case (DossierSimu)
```

Une fois le maillage créer, nous pouvons le visualiser dans ParaView :

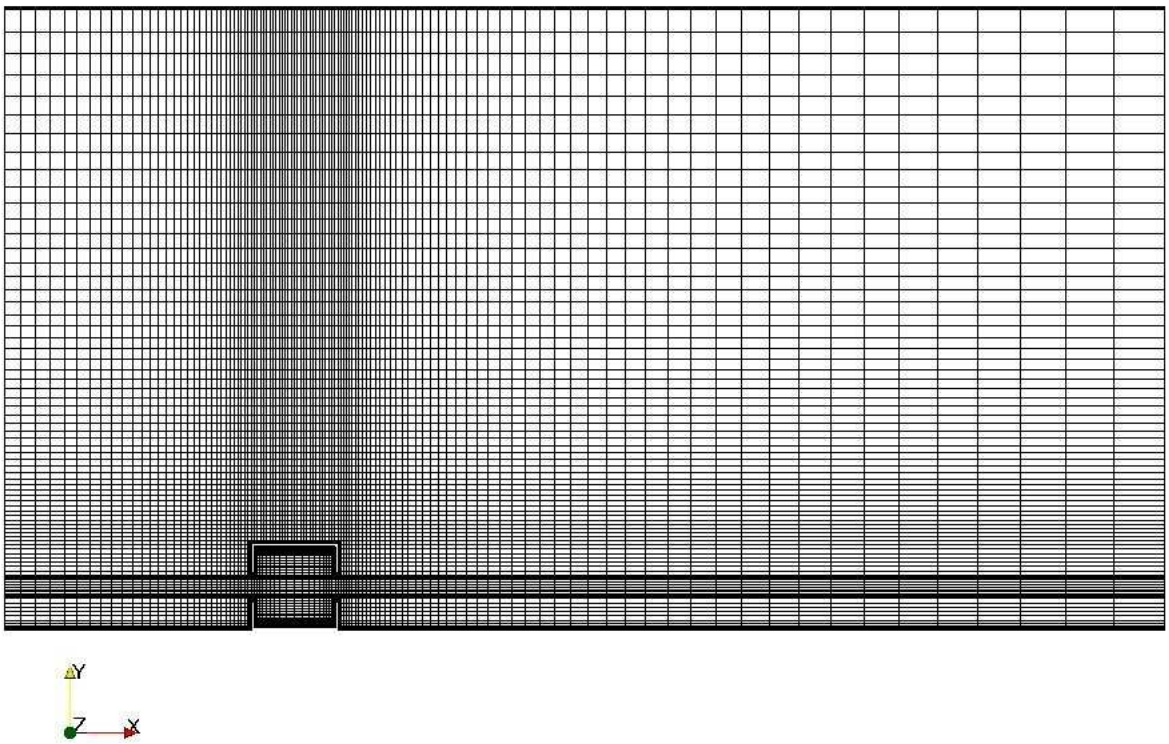


FIG. 4.11 – Maillage du domaine d'étude

4.5.4 Conclusion

A cause de la diffusion numérique, qui provient de l'erreur de discrétisation des termes de transport-convection des équations de Navier-Stokes, de transport des espèces et de l'énergie, le maillage de calcul doit être fin. Cette diffusion parasite est d'autant plus importante lorsque le maillage est plus grossier et entraîne des problèmes d'instabilités numériques.
Pour limiter la diffusion numérique, il faut utiliser un schéma de discrétisation précis (voir partie 4.7) et raffiner le maillage. Lorsque qu'il y avait des problèmes de convergence, j'ai donc dû adapter les paramètres vu précédemment

Le tableau suivant présente les statistiques du maillage utilisé pour la simulation :

Élément	Valeur
points	48 626
blocs (*blocks*)	63
faces	96052
faces internes	47426
cellules (*cells*)	23913
(*boundary patches*)	6

TAB. 4.2 – Statistique du maillage

Le maillage étant construit, il faut maintenant le configurer pour qu'il respecte les échelles de turbulence pour la LES. En effet, ces échelles de turbulence, nous renseigneront sur les tailles minimums et maximums des cellules du maillage et sur la configuration du système de résolution LES.

4.6 Détermination des échelles de turbulence pour la simulation LES

4.6.1 Introduction

Les échelles turbulentes sont déterminées à partir de l'énergie cinétique turbulente k et de la dissipation de cette énergie ε. Nous utiliserons donc le maillage précédent un peu plus grossier et un modèle de résolution RNG-k-epsilon pour un écoulement incompressible en régime stationnaire. Le solveur OpenFOAM utilisé est `simpleFoam`.

4.6.2 Modèle RNG-k-epsilon

Ce modèle utilise la modélisation statistique de la turbulence qui permet de calculer des grandeurs moyennes d'un écoulement turbulent, où toutes les échelles de la turbulence sont modélisées.

Les équations de Navier-Stokes moyennées sont les suivantes :

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\overline{u}_j \overline{u}_i) = -\frac{1}{\rho}\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}(2\nu \overline{S}_{ij} - \overline{u}'_j \overline{u}'_i) \qquad , i = 1,2,3 \tag{4.3}$$

Le problème de fermeture de ces équations a été largement étudié ces cinquante dernières années, mais le modèle statistique qui est sans aucun doute le plus répandu est le modèle de Launder et al. souvent désigné par "le modèle $k-\varepsilon$ ". Ce modèle fait l'hypothèse de viscosité turbulente, et s'appuie sur la détermination d'une échelle de longueur et une échelle de vitesse en résolvant deux équations de transport : une pour l'énergie cinétique turbulente k et une autre pour la dissipation d'énergie cinétique turbulente ε définis par :

$$k = \frac{1}{2}\overline{u}'_i \overline{u}'_i \tag{4.4}$$

$$\varepsilon = \nu \left(\frac{\partial \overline{u}'_i}{\partial x_i} \frac{\partial \overline{u}'_i}{\partial x_i} \right) \tag{4.5}$$

et dont les équations de transport s'écrivent sous la forme (pour i=1,2,3) :

$$\frac{\partial k}{\partial t} + \overline{u}_i \frac{\partial k}{\partial x_i} = \frac{\partial}{\partial x_i}\left((\nu + \frac{\nu_t}{\sigma_k}) \frac{\partial k}{\partial x_i} \right) + 2\nu_t \overline{S}_{ij} \frac{\partial \overline{u}_i}{\partial x_j} - \varepsilon \tag{4.6}$$

$$\frac{\partial \varepsilon}{\partial t} + \overline{u}_i \frac{\partial \varepsilon}{\partial x_i} = \frac{\partial}{\partial x_i}\left((\nu + \frac{\nu_t}{\sigma_\varepsilon}) \frac{\partial \varepsilon}{\partial x_i} \right) + c_{1_\varepsilon} \frac{\varepsilon}{k}\left(2\nu_t \overline{S}_{ij} \frac{\partial \overline{u}_i}{\partial x_j} \right) - c_{2_\varepsilon} \frac{\varepsilon^2}{k} \tag{4.7}$$

avec : $c_{1_\varepsilon} \simeq 1,42$, $c_{2_\varepsilon} \simeq 1,68$, $\sigma_\varepsilon \simeq 1$ et $\sigma_\varepsilon \simeq 1,39$.

La viscosité turbulente est alors évaluée par la relation suivante :

$$\nu_t = C_\mu \frac{\varepsilon^2}{k} \tag{4.8}$$

où $C_\mu \simeq 0,0845$.

4.6.3 Paramètres k et ε

Dans un système en coordonnée cartésienne, l'énergie de turbulence k est obtenue par la relation suivante :

$$k = \frac{1}{2}\left(U_x'^2 + U_y'^2 + U_z'^2 \right) \tag{4.9}$$

avec :

- C_μ= 0,0845,
- U'_x, U'_y, U'_z sont les fluctuations des composantes de la vitesse. Dans ce modèle, $U_x'^2 = U_y'^2 = U_z'^2 = \frac{5}{100} 1$ m.s^{-1}.

d'où :

$$k = \frac{3}{2}\left(\frac{5}{100}\right)^2 = 3,75 \times 10^{-3}\mathrm{m}^2\mathrm{s}^{-2} \tag{4.10}$$

et sa dissipation ε par la relation suivante :

$$\varepsilon = \frac{C_\mu^{0,75} k^{1,5}}{l} \tag{4.11}$$

avec : l est égale à 20% de la longueur caractéristique du système. l_c = 3m dans notre système.

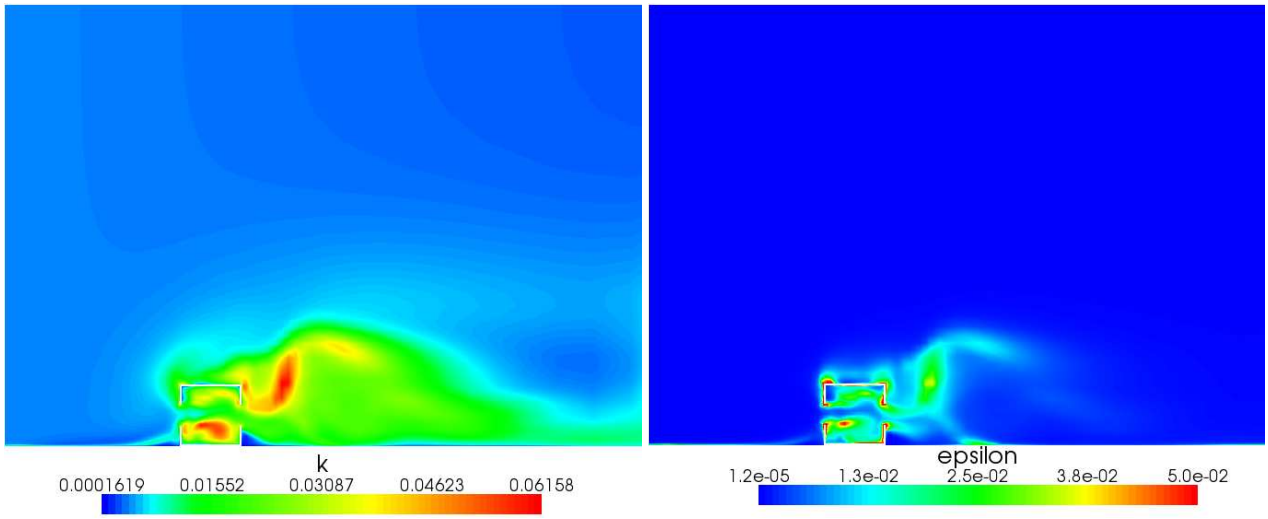
d'où :

$$\varepsilon = \frac{0,09^{0,75} 0,00375^{1,5}}{3} \approx 1,20 \times 10^{-5}\mathrm{m}^2\mathrm{s}^{-3} \tag{4.12}$$

Les valeurs obtenues des paramètres k et ε sont les valeurs initiales à t = 0.

4.6.4 Résultats de la simulation

Les résultats après 4000 itérations sont les suivants :



(a) Énergie cinétique turbulente k

(b) Dissipation de l'énergie ε

FIG. 4.12 – Paramètres k et ε

Les paramètres k et ε étant déterminés, pour tous le domaine, nous pouvons déterminer les échelles de turbulence.

4.6.5 Échelles de turbulence

La notion d'échelle est un instrument très utilisée en mécanique des fluides. Elle permet de cataloguer les différentes structures qui coexistent dans un écoulement et de leur donner une

certaine importance au sein de l'écoulement.
Un écoulement turbulent est composé de tout un ensemble de structures d'échelles très variées et peut être catalogué en deux catégories : les grandes échelles et les échelles dissipatives.

4.6.5.1 Les grandes échelles

Les échelles de longueur sont définies par la relation suivante :

$$L = \frac{k^{3/2}}{\varepsilon} \tag{4.13}$$

et les échelles de temps par la relation suivante :

$$T = \frac{k}{\varepsilon} \tag{4.14}$$

Ces deux échelles permettent de fixer les limites supérieures pour la taille des structures turbulentes.

4.6.5.2 Les échelles dissipatives

Au cours du processus de cascade d'énergie, les structures des échelles de longueur, se font de plus en plus petites, et le nombre de Reynolds associé aux structures diminue et lorsqu'il est faible (de l'ordre de l'unité), les effets visqueux deviennent prépondérants. Cependant, il existe une limite inférieure pour la taille des structures turbulentes.

Dans cette partie, on se propose de donner les expressions des échelles de temps τ, de vitesse u et de longueur η, caractéristiques pour les plus petites structures dissipativent qui peuvent se développer dans un écoulement. On obtient la relation suivante :

$$\eta = u\tau \tag{4.15}$$

et on constate que la viscosité cinématique ν joue pleinement son rôle si le nombre de Reynolds est de l'ordre de l'unité :

$$Re = \frac{\eta u}{\nu} \sim 1 \tag{4.16}$$

on admet que le taux de dissipation ε est proportionnel à l'inverse du temps τ :

$$\varepsilon = \frac{\nu}{\tau^2} \tag{4.17}$$

En combinant les relations (4.16), (4.17), (4.15), on obtient les échelles des structures dissipatives :

$$\tau = \left(\frac{\nu}{\varepsilon}\right)^{1/2}, \eta = \left(\frac{\nu^3}{\varepsilon}\right)^{1/4}, u = (\nu\varepsilon)^{1/4} \tag{4.18}$$

avec :

- τ une échelle de temps,
- η une échelle de longueur.

Ces échelles sont appelées aussi **les échelles de Kolmogorov**[1].

Il existe aussi les **échelles de Taylor** qui sont souvent utilisées dans les études d'écoulement homogène et isotrope. L'échelle de longueur λ et l'échelle de temps λ_t sont définies à partir des relations suivantes :

$$\lambda = \left(\frac{10\nu k}{\varepsilon}\right)^{1/2}, \lambda_t = \left(\frac{15\nu}{\varepsilon}\right)^{1/2} \tag{4.19}$$

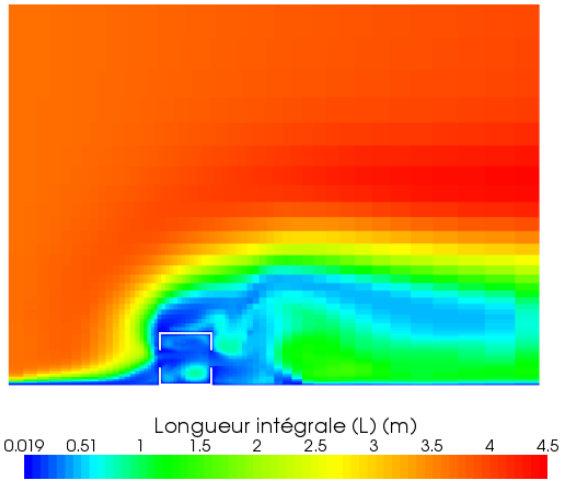
J'ai réalisé un programme, en langage C++, qui permet à partir de la simulation RNG-k-ε, de calculer toutes les échelles. Ce calcul est très utile pour adapter mon maillage à la LES (tailles des mailles) et pour l'initialisation du solveur (pas de temps, durée de la simulation,...).

```
echelles : scalaire ;
i : réel ;

[Les échelles turbulentes sont initialisées]
L ← 0 ;
T ← 0 ;
η ← 0 ;
τ ← 0 ;
λ ← 0 ;
λt ← 0 ;

Pour (i=TpsDebutSimu ; i=TpsFinSimu ; i++) faire
    [Lecture de k et de ε]

    Si (k, ε existe) Alors
        calculs de toutes les échelles ;
        NbrVectU=NbrVectU+1 ;
    Fin Si
Fin Pour
[Évaluation du max et du min]
max(échelles) et min(écheles) ;
[Écriture des échelles]
échelles → dossier ;
```

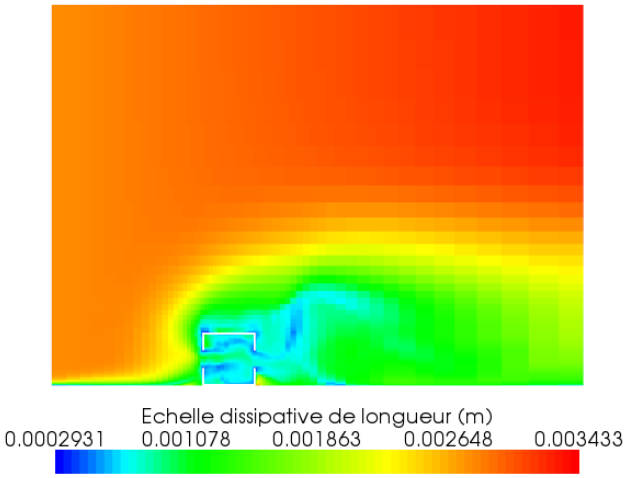
Algorithme 1: Calcul des échelles turbulentes

[1]Ces échelles sont déduites des spectres d'énergie cinétique et de la dissipation de la chaleur.

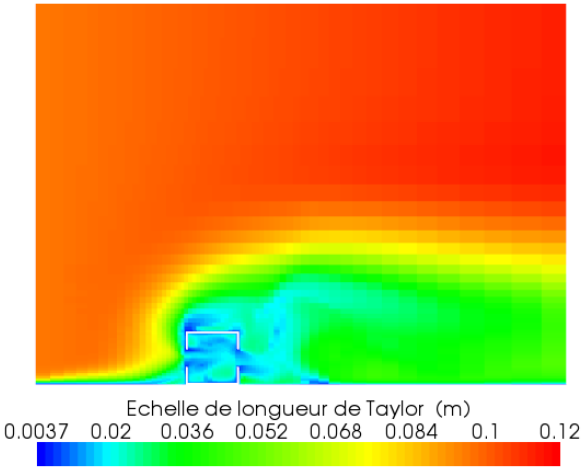
Les différentes illusrations et tableau suivant présentent les résultats obtenus des échelles, après la simulation RNG-k-ε :



(a) Longueur intégrale L



(b) Echelles de longueur η



(c) Echelles de longueur λ

FIG. 4.13 – Echelles turbulentes

Échelle	Limite min	Limite max
L	0,018 m	4,45 m
η	0,00029 m	0,0034 m
λ	0,0037 s	0,116 m
T	0,001 s	86 s
τ	0,001 s	0,75 s
λ_t	0,001 s	2,92 s

TAB. 4.3 – Échelles obtenues pour le maillage

Les échelles de turbulence étant définies, nous pouvons réaliser notre simulation LES.

4.7 Application de la LES à la convection turbulente sur un bâtiment ouvert en 2D

4.7.1 Introduction

Le but de cette partie, est d'évaluer le comportement d'un écoulement turbulent, en régime instationnaire, utilisant la LES avec un modèle de Smagorinsky.

Dans un premier temps, nous expliquerons l'algorithme de calcul des équations de conservations. Puis, dans un second temps, nous définirons les différents domaines à paramétrer et les choix numériques qui ont été retenus. Enfin, nous essayerons d'interpréter les résultats obtenus.

4.7.2 Algorithme de calcul

4.7.2.1 Introduction

Le problème principal des équations de Navier Stokes incompressibles est qu'il n'existe pas d'équation pour la pression, séparée de celle de la quantité de mouvement. De plus, la vérification de l'équation de continuité n'est pas garantie pour un champ de vitesse qui vérifie l'équation de quantité de mouvement.

OpenFOAM utilise un solveur découplé (`oodles`), c'est à dire que les équations qui gouvernent l'écoulement sont résolues séquentiellement (découplées l'une de l'autre). Un calcul itératif est nécessaire pour des équations non linéaires et interdépendantes.

4.7.2.2 Explication de la méthode

Partons de l'équation de continuité :

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{4.20}$$

et de l'équation de conservation de mouvement :

$$\rho \frac{\partial u_i}{\partial t} = -\frac{\partial p}{\partial x_i} + H_i \tag{4.21}$$

avec : $H_i = -\rho \frac{\partial u_i u_j}{\partial x_j} + \frac{\partial \sigma_{ij}}{\partial x_j}$

La démarche la plus simple pour les équations instationnaires est d'effectuer un avancement en temps de manière explicite :

$$u_i^{n+1} = u_i^n + \frac{\Delta t}{\rho}(H_i^n - \frac{p^n}{x_i}) \tag{4.22}$$

On suppose que la continuité est vérifiée au temps n, alors $\frac{\partial u_i^n}{\partial x_i} = 0$, et on cherche à faire en sorte qu'elle soit vérifiée au temps $n+1$, soit $\frac{\partial u_i^{n+1}}{\partial x_i} = 0$.

Alors, il important de vérifier l'expression suivante :

$$\frac{\partial}{\partial x_i}\left(\frac{\partial p^n}{\partial x_i}\right) = \frac{\partial H_i^n}{\partial x_i} \tag{4.23}$$

Chaque itération est constituée des étapes suivantes :

1. on prend un champ de vitesse u_i^n au temps n, à divergence nulle,
2. on calcule H_i^n et sa divergence $\frac{\partial H_i^n}{\partial x_i}$,
3. on résout l'équation de Poisson pour p^n,
4. on calcule u^{n+1} en fonction de : u_i^n, H_i^n et de p^n, par un schéma explicite. Alors, on a la garantie que u^{n+1} est bien à divergence nulle, et on recommence ainsi de suite.

OpenFOAM utilise des méthodes implicites, c'est-à-dire pour lesquelles le champ de vitesse u^{n+1} est une fonction à la fois de données au temps n et de données au temps $n+1$. Elles sont plus rapides que les méthodes explicites.

4.7.2.3 Méthodes de type "prédiction-correction"

L'algorithme couramment utilisé pour effectuer l'avancement en temps est appelé **SIMPLE** : **S**emi-**I**mplicit **M**ethod for **P**ressure **L**inked **E**quation (1984).

Il ne s'agit pas d'une méthode implicite pure, qui consiste à effectuer des inversions de matrices, mais d'une méthode de projection-correction, dans laquelle, on estime la solution par des quantités intermédiaires, puis on les corrige afin qu'elles vérifient bien la divergence nulle, et ainsi de suite jusqu'à ce que les corrections appliquées tendent vers zéro. Il s'agit d'un traitement pseudo-instationnaire des équations pour converger vers une solution stationnaire. Les étapes sont les suivantes :

1. au départ on connaît u_i^n et p^n au temps t_n,
2. on résout de manière itérative les équations de quantité de mouvement linéarisées et on obtient un champ intermédiaire u_i^{m*} . La correction de vitesse est définie par $u_i^m = u_i^{m*} + u'$,
3. on introduit une correction de pression, telle que $p^m = p^{m-1} + p'$ et on résout l'équation pour de correction de pression p',
4. on retourne à l'étape 1 et on effectue la sous-itération $m+1$, ainsi de suite jusqu'à ce que p' et u' soient négligeables,
5. alors, on obtient $u_i^{n+1} = p^m$ et $p^{n+1} = p^m$.

Pour notre simulation LES, nous utilisons un autre algorithme qui améliore la rapidité de convergence. Il s'agit de l'algorithme **PISO**.

L'algorithme PISO (**P**ressure **I**mplicit with **S**plitting of **O**perators (1966)) utilise une première correction de pression identique à celle de SIMPLE. Une deuxième étape de correction est effectuée (au cours de la même itération) en vue de mieux satisfaire la conservation de quantité de mouvement et l'équation de continuité (voir code du solveur de OpenFOAM en annexe).

4.7.3 Discrétisations

Rappel :

$$\frac{\partial u}{\partial t} + \nabla.(uu) = -\frac{1}{\rho}\nabla p + \nabla.\left[(\nu + \nu_t)\nabla u\right] \tag{4.24}$$

Dans ce que va suivre, je n'expliquerais que la résolution du terme temporel et du terme convectif. Cependant, j'énoncerais la discrétisation du terme de sous-maille sans rentrer dans les détails.

4.7.3.1 Terme temporel : $\partial u/\partial t$

Pour résoudre ce terme, j'utilise un schéma de discrétisation appelé **Crank Nicholson** plus restrictif que les schémas d'**Euler**. En effet, les schémas d'Euler sont d'ordre 1 en temps et le schéma Crank Nicholson est d'ordre 2 en temps et en espace. Il est obtenu en faisant une moyenne des schémas d'Euler progressif (schéma explicite) et rétrograde (schéma implicite).

Afin d'assurer la stabilité du schéma temporel, il ne faut pas que le nombre de courant Co soit supérieur à 1. Le nombre de Courant Co est la mise sous forme adimensionnelle du pas de discrétisation en temps utilisé pour la résolution numérique des équations différentielles en mécanique des fluides. Il s'exprime de la façon suivante :

$$Co = \frac{\partial t|u|}{\partial_x} \tag{4.25}$$

avec :

- ∂t est le pas de temps (s),
- ∂_x est le pas du maillage dans l'espace (m),
- u est la vitesse (m/s).

Le nombre de Courant est un indicateur adimensionnel de la valeur du pas de temps.

4.7.3.2 Terme convectif : $\nabla.(u.u)$

Un critère important de qualité des résultats est la discrétisation en espace des termes convectifs. La discrétisation de ce terme est en effet génératrice de diffusion numérique et d'instabilités du schéma. Les discrétisations centrées sont génératrices d'instabilités. La discrétisa-

tion décentrée amont simple est stable et robuste, mais extrêmement diffusive : elle introduit un coefficient de diffusion artificiel égale, en monodimensionnel, à :

$$D = \frac{|u|\partial_x}{2} \tag{4.26}$$

où $|u|$ est le module de la vitesse et ∂_x le pas en espace.

Le résultat est donc très sensible au raffinement du maillage : raffiner permet de diminuer la diffusion numérique. Les meilleures méthodes sont les méthodes décentrées amont d'ordre élevé, comme les méthodes de type "**QUICK**" ou les méthodes lagrangiennes.

Les méthodes de type QUICK (**Q**uadratic **U**pstream **I**nterpolation **S**chemes) sont souvent utilisées pour résoudreles termes de convection :

$$\frac{\partial u_i u_j}{\partial x_j} = \frac{(u_i u_j)^n_{j+1/2} - (u_i u_j)^n_{j-1/2}}{\partial x} \tag{4.27}$$

Ce type de schéma est basé sur l'utilisation d'une grille décentrée (staggered en anglais) où les quantités sont calculées de la façon suivante :

$$u_{j-1/2} = \frac{1}{2}(u_j + u_{j+1}) - \frac{q}{3}(u_{j-2} - 2u_{j-1} + u_{j-1}) \tag{4.28}$$

$$u_{j+1/2} = \frac{1}{2}(u_j + u_{j+1}) - \frac{q}{3}(u_{j-1} - 2u_j + u_{j+1}) \tag{4.29}$$

avec q = 0,375 (Léonard, 1979).

Cependant, l'interpolation quadratique QUICK, pour notre problème 2D sous OpenFOAM, n'a pas fonctionné pour cause d'instabilité. Suite à une lecture très récente d'une **H**abilitation à **D**iriger les **R**echerches (HDR), de Patrice JOUBERT [7], une remarque pertinente (page 114), nous apprenons que pour stabiliser la solution, il faut utiliser un limiteur de flux qui permet d'éviter des oscillations "parasites" qui pourraient se produire lors d'une résolution (mot clé OpenFOAM :`limitedLinear`). L'utilisation de ce limiteur a permis la convergence de notre système.

4.7.3.3 Terme de sous-maille : $\nabla.[(\nu + \nu_t)\nabla u]$

Le modèle utilisé, pour modéliser la dissipation des petites échelle,s est de type Smagorinsky. Le modèle sous-maille calcule le comportement des grandes échelles, en modélisant l'action des petites échelles. La modélisation se fait en modélisant l'action dissipative des petites échelles sur les grandes échelles, au moyen d'une viscosité tourbillonnaire. Dans notre étude, nous utilisons, un modèle plus précis : un **modèles dynamique**. Il permet d'obtenir une meilleure estimation des constantes intervenant dans la modélisation de type Smagorinsky. Le modèle dynamique est le modèle de Germano M. & al. (1990) [6]. L'idée de ce modèle est d'appliquer un double filtrage.

Près des parois, il est couplé avec un modèle de lois de parois de type Van Driest.

4.7.4 Récapitulatif

ordinateur	Intel Pentium 4, CPU 3GHz, RAM 2Gio
solveur LES	oodles, incompressible, newtonien
durée simulation	100 s
pas de temps	0,005 s
modèle de sous-maille	Smagorinsky dynamique
lois de parois	Van Driest
boucle de prédiction-correction	PISO

TAB. 4.4 – Configuration du fichier de contrôle

La configuration d'une simulation LES, demande un travail amont très rigoureux. Afin de pouvoir optimiser ce travail, il était nécessaire d'établir des outils informatiques où j'ai consacré énormément de temps (assimilation au concept de la LES, comprendre les langages de programmations, ...).

4.8 Outils informatiques créés

4.8.1 Moyennes temporelles

A partir des outils de traitements de données d'OpenFOAM, j'ai élaboré un programme, en langage C++, qui permet de calculer la moyenne temporelle des vitesses(u). Ce calcul de moyenne filtre le signal et ramène le problème à un écoulement stationnaire.

```
VectUmoy : vecteur ;
VectU : vecteur ;
i : réel ;
NbrVectU : entier ;

[VectUmoy est initialisé]
VectMoy ← (0,0,0) ;
Pour (i=TpsDebutSimu ; i=TpsFinSimu ; i++) faire
    [Lecture du vecteur vitesse]
    VectU ← (VectUx,VectUy,VectUz)
    Si (VectU existe) Alors
        VectUmoy=VectUmoy+VectU ;
        NbrVectU=NbrVectU+1 ;
    Fin Si
Fin Pour
[Calcul de la moyenne]
VectUmoy=VectUmoy/NbrVectU ;
[Écriture du vecteur vitesse moyen]
VectUmoy → dossier ;
```

Algorithme 2: Calcul du vecteur vitesse moyen

Après chaque modification du code C++, une exécution de la commande Bash `wmake` est nécessaire. Celle-ci permet de compiler le code afin que les modifications soient prise en compte. Ces outils de calcul ne peuvent être exécutés qu'à la fin de chaque simulation. La commande Bash pour l'exécution est la suivante :

```
(NomOutil) -case (DossierSimu)
```

L'avantage d'utiliser les librairies C++ d'OpenFOAM, pour créer mes outils de calcul, se trouve dans la simplicité de visualisation des données obtenues avec ParaView.

4.8.2 Traitements des données

La fonction `Probes` d'OpenFOAM permet de réaliser des relevés de données, pendant la simulation, sur des grandeurs caractéristiques de l'écoulement (vitesse, pression, ...) pour chaque pas de temps et pour une localisation donnée. Par exemple, j'ai placé une série de probes à l'entrée et à la sortie du bâtiment (42 points de mesures), afin d'évaluer p et u. L'inconvénient est que les fichiers de sortie ne sont pas pris en compte dans ParaView. Il est donc nécessaire d'effectuer un traitement sur les fichiers de sortie (`.txt`). Pour cela, j'ai réalisé deux programmes dans deux langages différents : Shell (Bash) et Python. Le choix des langages est basé notamment sur la rapidité de mise en place pour atteindre l'objectif fixé. Le traitement se décompose en deux étapes :

4.8.2.1 Mise en page des données

Les fichiers textes, issus de la fonction `Probes`, ont un formalisme spéciales difficiles à traiter par un tableur quelconque. Il faut donc formater ces fichiers afin de pouvoir les préparer pour les exploiter (graphiques, ...). Une solution rapide et simple est d'utiliser un script Bash et notamment la commande `sed` qui permet de remplacer une chaîne de caractères (".",",", ...) ou une expression régulière (\t : tabulation, \n : retour à la ligne, ...) par une autre. La syntaxe de la commande est la suivante :

```
sed -e 's/caractère à remplacer}/caractère de remplacement/g' f.txt
```

#	Probe1	Probe2	...
#	x1	x2	...
#	y1	y2	...
#	z1	z2	...
# Time			
0.005	(2.392,0.074,0)	(2.160,0.051,0)	...
0.01	(2.87,0.047,0)	(2.135,0.079,0)	...
⋮	⋮	⋮	...

TAB. 4.5 – Fichier de données généré par OpenFOAM

La mise en page souhaitée est la suivante :

	Probe1 (x1,y1,z1)			Probe2 (x2,y2,z2)			...
t(s)	Ux	Uy	Uz	Ux	Uy	Uz	...
0.005	2.392	0.074	0	2.160	0.051	0	...
0.01	2.87	0.047	0	2.135	0.079	0	...
⋮	⋮	⋮	⋮	⋮	⋮	⋮	...

TAB. 4.6 – Mise en forme des données

Voici l'algorithme mis en place pour résoudre ce problème :

```
file : chaîne de caractères ;

Si ($@ = 0 → vrai) Alors
    [Test d'erreurs]
    echo "ERREUR ! ! ! Exemple : ./format.sh fichier1.txt fichier2.txt"
    exit
Sinon
    Pour file dans "$@" faire
        sed -e 's/(//g' $file > "$file".tmp && mv -f "$file".tmp "$file"
        [On répète l'opération pour atteindre notre objectif de mise en forme]
        sed -e ...
        ⋮

    Fin Pour
Fin Si
[Appel du programme Python]
traitement.py
```

Algorithme 3: Mise en forme des fichiers de sortie

Comme on peut le constaté, dans l'algorithme précédent, le script appelle mon deuxième programme écrit en langage Python.

4.8.2.2 Exploitation des données

Le langage utilisé pour cette application est Python. Munit de librairies puissantes, qui reprennent le "meilleur" de chaque langage, Python permet directement d'atteindre mon objectif qui est d'exploiter mes données (calculs, graphiques, ...). Pour être le plus efficace, j'ai utilisé les librairies suivantes :

- Math (librairie mathématique),
- Pylab (équivalent au shareware MatLab),
- MatPlotLib (graphiques).

J'ai donc à partir du fichier généré par mon programme bash, extrait les données notamment la composante du vecteur u suivant l'axe $\vec{x}$, la pression p, pour chaque pas de temps. Mon programme permet de calculer des vitesses moyennes, des débits et de mettre les données sous formes de graphique afin de pouvoir les exploiter. Les graphiques sont sauvegardés et transformés en images (`.eps`).

4.9 Résultats et discussions

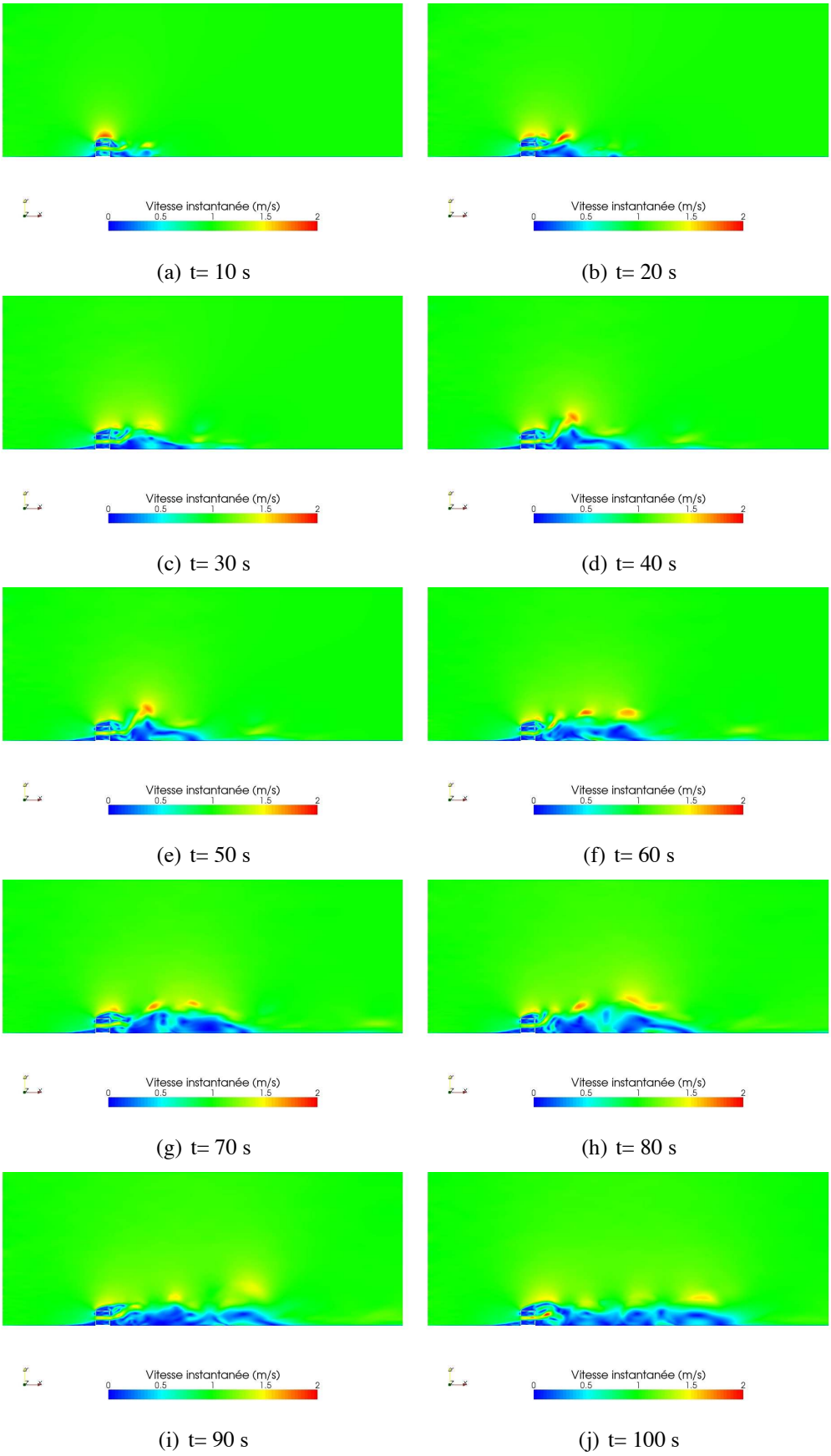
4.9.1 Introduction

Les simulations LES sont très coûteuses en temps de calcul. Le laboratoire LPBS est munit d'un calculateur munit de 32 processeurs. Il nous a permis de simuler notre étude sur des cas tests, notamment en 3D, avec différent type de mailles (non explicité dans ce rapport). Pour différents problèmes (complexité du maillage, stabilité du système, effets de bord, ... ;), ces cas tests, nous ont amenés à la conclusion d'utiliser un maillage 2D, expliqué précédemment. Le temps de calcul, pour une simulation de 100 s d'écoulement 2D, sur mon Pentium 4, est d'environ 10 h. Ce temps de calcul n'est pas essentiellement dû à la LES. Les relevés de p et de U, avec la fonction `probes`, sur l'entrée et la sortie du bâtiment ralentissent le temps de calcul. D'ailleurs, les résultats présentés, dans la partie suivante, dépendent de ces relevés.

L'objectif essentiel de ce stage était de mettre en place des outils informatiques pour la LES. Les résultats suivants nous permettent en quelques sortes de validé nos outils.

4.9.2 Champ de vitesse instantanée

ParaView permet d'observer l'écoulement autour du bâtiment comme le montre les illustrations suivantes :



(a) t= 10 s (b) t= 20 s

(c) t= 30 s (d) t= 40 s

(e) t= 50 s (f) t= 60 s

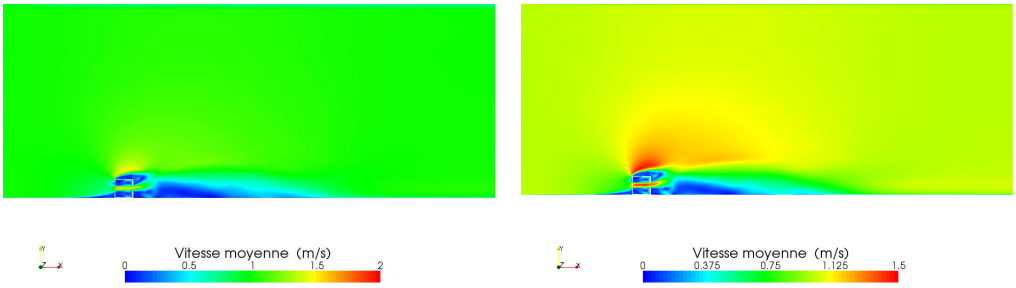
(g) t= 70 s (h) t= 80 s

(i) t= 90 s (j) t= 100 s

FIG. 4.14 – Champ instantanée de la vitesse

4.9.3 Champ de vitesse moyennée

Le programme mis en place précédement (partie 4.8.1) permet de filtrer le champ de vitesse instannée, pour ce ramener à un cas stationnaire.



(a) échelle identique au champ instantanée

(b) échelle adaptée

FIG. 4.15 – Champ moyenné de la vitesse

4.9.4 Etude de l'écoulement

4.9.4.1 Introduction

La notion de rotationnel de la vitesse est essentielle en mécanique des fluides. Elle décrit une rotation de la particule fluide. Si l'écoulement est irrotationnel (son rotationnel est nul en tout point, l'équation 4.30 est nulle), les vitesses dérivent d'un potentiel. En termes mathématiques, le vecteur vitesse est alors le gradient du potentiel. Si le fluide peut être considéré comme incompressible, la divergence de ce vecteur s'annule.

On appel vecteur tourbillon :

$$\vec{\Omega} = \frac{1}{2}\vec{rot}\vec{u} \tag{4.30}$$

4.9.4.2 Type d'écoulement

L'outil de post-processing `flowType` , permet de connaître la nature de l'écoulement (rotationel ou non), pour chaque pas de temps. Je l'ai modifié, de sorte qu'il me définisse le type d'écoulement moyen. Il est défini, de la façon suivante :

$$myFlowType = \frac{|\Gamma| - |\Theta|}{|\Gamma| + |\Theta|} \tag{4.31}$$

avec : $\Gamma = \frac{1}{2}(grad(VectUmoy) + (grad(VectUmoy))^T)$

et : $\Theta = \frac{1}{2}(grad(VectUmoy) - (grad(VectUmoy))^T)$

si :

– *myFlowType*=-1 alors l'écoulement est rotationnel,
– *myFlowType*=0 alors l'écoulement est cisaillé,
– *myFlowType*=1 alors l'écoulement est irrotationnel.

D'après les résultats de simulation, figure 4.16, on constate que l'écoulement est irrotationnel dans tout l'espace sauf en un certain nombre de points singuliers pour lesquels le vecteur tourbillon est différent de zéro. Ces points se groupent suivant un certain nombre de filets tourbillons distincts (dans le bâtiment, en bleu). Cet écoulement est dit : **écoulements à potentiel des vitesses avec circulation**.

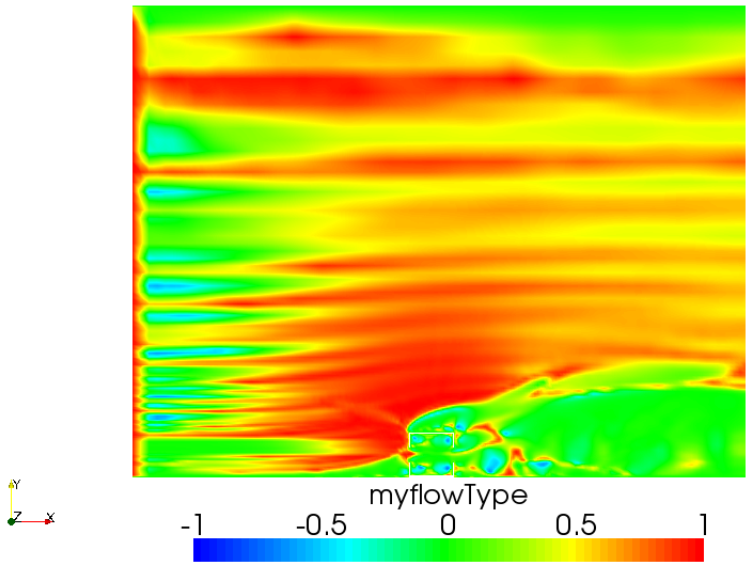


FIG. 4.16 – Type d'écoulement

4.9.5 Débit massique

4.9.5.1 Evolution du débit massique à l'entrée et à la sortie du bâtiment

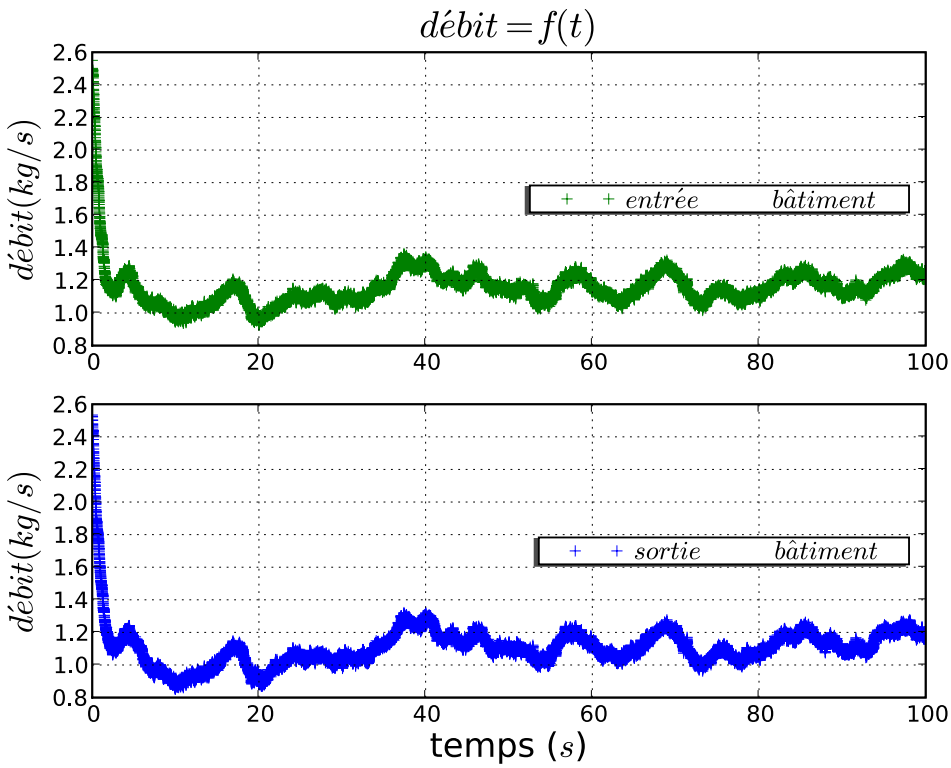


FIG. 4.17 – Débit en fonction du temps

Le débit est définie par la relation suivante :

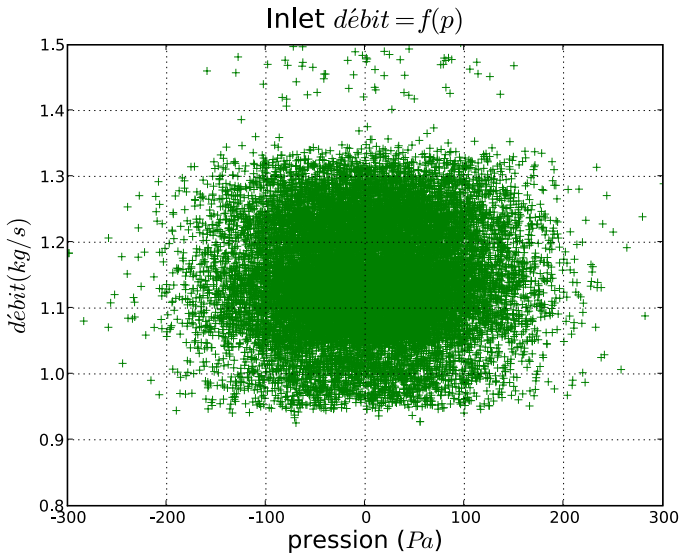
$$\dot{m} = \rho U_{moy}.S \tag{4.32}$$

avec :

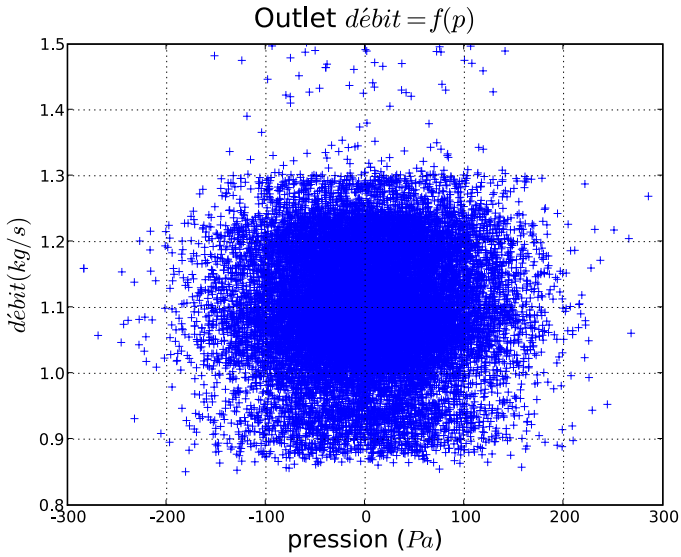
- U_{moy} est la moyenne des vitesses déterminées à partir des 21 "probes" placés à l'entrée et à la sortie. Les relevés sont équidistants entre eux.
- S égale à $1m^2$.

On constate que le débit, après 20 s, oscille entre 1 et 1,4 kg/s.

4.9.5.2 Evolution du débit massique en fonction de la pression, à l'entrée et à la sortie du bâtiment



(a) Entrée du bâtiment



(b) Sortie du bâtiment

FIG. 4.18 – Evolution du débit

Conclusion générale et perspectives

Dans le travail présenté, nous avons suivi deux directions de recherche : la compréhension de la turbulence avec ses phénomènes physiques et numériques mis en jeu ; et le développement d'outils informatiques pour la simulation des écoulements numériques.

Dans les deux premiers chapitres, nous avons exposé les approches utilisés dans l'étude de la turbulence et nous avons présenté en détail la méthode de simulation des grandes échelles (LES) afin de mettre en évidence les critères essentiels (échelles turbulentes, modèle de sous-mailles, ...) pour paramétrer notre simulation d'un écoulement autour d'un bâtiment ouvert.

Dans le troisième chapitre, on présente en détails les outils mis en oeuvre pour la simulation LES. Le code de calcul OpenFOAM et les autres programmes annexes (traitement des données, outils de calcul, ...), ont permis de mener à bien notre simulation LES. L'avantage d'avoir utilisé un code Open Source, nous a aidé à mieux assimiler la complexité de la résolution numérique des équations de Navier-Stokes et à permettre de modifier le code afin d'établir de nouveaux critères plus pertinents que ce proposé par OpenFOAM.

La LES reste une méthode de résolution coûteuse en temps de calcul. Une des perspectives de cette étude, serait de simplifier les modèles utilisés, pour ce type d'étude. L'idée serait d'établir des zones autour du bâtiment où le tenseur de Reynolds, responsable des fluctuations dans l'écoulement, joue un rôle important et de proposer des modèles numériques qui remplacerait ce tenseur par autre plus simple. Cette méthode pourait considérablement diminuer le temps de calcul. Une étude expérimentale serait nécessaire pour valider les modèles proposés par cette perspectives.

La formation exigeante, du master 2me ISI GST de Pau, que j'ai reçu, ma permis de travailler directement lors de mon arrivé au LPBS. Cependant, il a était nécessaire de me remettre à niveau dans le domaine de la mécanique des fluides, notamment dans le domaine de la turbulence, qui n'avait pas était abordée en détail dans mon cursus scolaire. Cependant, le fait de côtoyer régulièrement des thésards et des post-doc m'a amélioré dans la compréhension des phénomènes mis en jeu dans la mécanique des fluides. De plus, j'ai énormément enrichie mon expérience dans ce domaine et plus précisément dans la modélisation numérique où en six mois, j'ai appris plusieurs langage de programmation : shell, C++, python, batch.

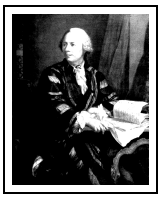
Le fait que ce soit le deuxième stage, dans un laboratoire de recherche, ma éclairé sur mon avenir professionnel. Ce stage renforce mes convictions à continuer mes études. Je commence à prospecter pour une demande de thèse dans le secteur de l'énergie et plus particulièrement dans le développement de modèles numériques.

Annexe A

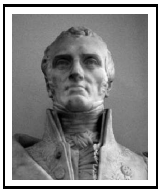
ANNEXES

Un peu d'histoire

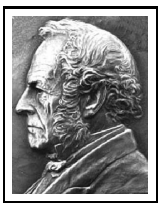
Présentation de quelques "Grands Hommes de sciences" ayant travaillés dans le domaine des milieux continus.



Leonhard EULER
Mathématicien suisse.
Né à Bâle le 15 avril 1707 et mort Saint-Pétersbourg le 18 septembre 1783.
Euler fut l'élève de Jean Bernoulli à l'université de Bâle. Ami de Daniel Bernoulli, il rejoignit celui-ci à l'Académie de Saint-Pétersbourg en 1727.
Euler fut le grand calculateur du XVIIIe siècle (théorie des équations différentielles, variables complexes, etc) commençant à donner aux mathématiques leur forme actuelle (notations $f(x)$, e, i, π, etc).
Dans son ouvrage "Mechanica" (1736-37), il présente la dynamique newtonienne sous forme analytique et non plus sous forme géométrique. C'est Euler qui, concevant la pression au sein d'un fluide comme grandeur locale, obtint l'équation du mouvement qui porte son nom. En l'intégrant, il écrivit l'équation dite "de Bernoulli" qu'on ne trouve pas de façon aussi explicite dans l'ouvrage majeur sur ce sujet "Hydrodynamica".



Claude Louis Marie Henri NAVIER
Ingénieur et scientifique français.
Né à Dijon le 10 février 1785 et mort à Paris le 21 août 1835.
Diplôme de l'École polytechnique et l'École des ponts et chaussées, Navier enseigna dans ces deux institutions. Ses travaux d'ingénieur portèrent notamment sur la construction de ponts. En 1824, il fut élu à l'Académie des sciences.
Sur le plan théorique, Navier établit, en 1821, les équations fondamentales de l'élasticité. En mécanique des fluides, Navier proposa les équations dites de "Navier-Stokes" pour les écoulements d'un fluide incompressible.



George Gabriel STOKES
Mathématicien et physicien irlandais.
Né à Skreen le 13 août 1819 et mort à Cambridge le 1er février 1903.
Élève de Francis Newman au bristol College, Stokes y fut remarqué par ses aptitudes en mathématiques. Il devint professeur dans cette discipline à Cambridge en 1849. Élu membre de la Société Royale en 1854, il en fut le secrétaire pendant 30 ans, puis président pendant 5 ans.
Stokes exerça principalement son activité scientifique en hydrodynamique, en élasticité et en optique. De 1842 à 1845, Stokes publia des articles importants sur les mouvements d'un fluide incompressible, les lois de la viscosité, et les mouvements d'un solide élastique. La loi dite "de Stokes" régit le mouvement des particules solides dans les fluides. En optique, il s'intéressa à la théorie ondulatoire de la lumière, ainsi qu'à la fluorescence.

Code C++ du solveur Oodles

```
/*-------------------------------------------------------------------------*\
  =========                 |
  \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
   \\    /   O peration     |
    \\  /    A nd           | Copyright (C) 1991-2008 OpenCFD Ltd.
     \\/     M anipulation  |
-------------------------------------------------------------------------------
Application
    oodles

Description
    Incompressible LES solver.

\*-------------------------------------------------------------------------*/

#include "fvCFD.H"
#include "incompressible/singlePhaseTransportModel/singlePhaseTransportModel.H"
#include "incompressible/transportModel/transportModel.H"
#include "incompressible/LESModel/LESModel.H"
#include "IFstream.H"
#include "OFstream.H"
#include "Random.H"

// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

int main(int argc, char *argv[])
{
    #include "setRootCase.H"
```

```
#include "createTime.H"
#include "createMeshNoClear.H"
#include "createFields.H"
#include "initContinuityErrs.H"

Info<< "\nStarting time loop\n" << endl;

for (runTime++; !runTime.end(); runTime++)
{
    Info<< "Time = " << runTime.timeName() << nl << endl;

    #include "readPISOControls.H"
    #include "CourantNo.H"

    sgsModel->correct();

    fvVectorMatrix UEqn
    (
        fvm::ddt(U)
      + fvm::div(phi, U)
      + sgsModel->divDevBeff(U)
    );

    if (momentumPredictor)
    {
        solve(UEqn == -fvc::grad(p));
    }

    // --- PISO loop
    for (int corr=0; corr<nCorr; corr++)
    {
        volScalarField rUA = 1.0/UEqn.A();

        U = rUA*UEqn.H();
        phi = (fvc::interpolate(U) & mesh.Sf())
            + fvc::ddtPhiCorr(rUA, U, phi);

        adjustPhi(phi, U, p);

        for (int nonOrth=0; nonOrth<=nNonOrthCorr; nonOrth++)
        {
            fvScalarMatrix pEqn
            (
                fvm::laplacian(rUA, p) == fvc::div(phi)
            );
```

```
                pEqn.setReference(pRefCell, pRefValue);

                if (corr == nCorr-1 && nonOrth == nNonOrthCorr)
                {
                    pEqn.solve(mesh.solver(p.name() + "Final"));
                }
                else
                {
                    pEqn.solve(mesh.solver(p.name()));
                }

                if (nonOrth == nNonOrthCorr)
                {
                    phi -= pEqn.flux();
                }
            }

            #include "continuityErrs.H"

            U -= rUA*fvc::grad(p);
            U.correctBoundaryConditions();
        }

        runTime.write();

        Info<< "ExecutionTime = " << runTime.elapsedCpuTime() << " s"
            << "  ClockTime = " << runTime.elapsedClockTime() << " s"
            << nl << endl;
    }

    Info<< "End\n" << endl;

    return(0);
}


// ************************************************************************* //
```

Bibliographie

[1] Olivier CADOT. *Introduction à la turbulence*, volume 2me année cycle Ingénieur-ENSTA. Paris. vii, 9

[2] Jean COIRIER. *Mécanique des milieux continus. Concept de base*, volume 2me cycle écoles d'ingénieurs. Paris, 1997. 4

[3] Alexandre CHATELAIN. *Simulation des Grandes Echelles d'écoulements turbulents avec transferts de chaleur*. PhD thesis, CEA Grenoble, 2004. 4

[4] S.DUBOS. *Simulation des grandes échelles d'écoulements turbulents supersoniques*. PhD thesis, Institut National des Sciences Appliquées de Rouen, 2005. 5

[5] Delia JIROVEANU. *Analyse théorique et numérique de certains modèles de viscosité turbulente*. PhD thesis, Université Joseph Fourier-Grenoble I, 2002. 7, 11, 15

[6] P. MOIN M. GERMANO, U. PIOMELLI. A dynamic subgrid-scale eddy viscosity model. *Fluids*, pages p–1760 p–1765, 1991. 13, 37

[7] Patrice JOUBERT. *Contribution à l'amélioration de la prévision des écoulements de convection naturelle turbulente en cavité*. La Rochelle, LEPTAB, 2008. 37

[8] Francis ALLARD Alain BASTIDE, Harry BOYER. Natural ventilation-a new method based on the walton model applied to cross-ventilated buildings having two large external openings. *Internationnal Journal of Ventilation*, 6 No 3 :p–195 p–202, 2007.

[9] J. BOUDET FANG and L. Shao. Les échanges inter-échelles en simulation des grandes échelles. In 18me *Congrès Français de Mécanique*, 27-31/08/2007.

Résumé

Titre du document / Document title :
Simulation des grandes échelles (SGE) d'écoulements de fluides incompressibles
Large-eddy simulation (LES) of incompressible fluid flows.

Auteur / Author
Boris BRANGEON

Affiliation de l' auteur / Author Affiliation
Université de PAU (64000)

Résumé / Abstract

Dans le cadre de la simulation des grandes échelles en mécanique des fluides, le travail réalisé au cours de stage a pour objectif :
– d'expliquer l'étude théorique et pratique de la simulation des grandes échelles,
– développer des outils informatiques, décrivant au mieux la physique de la turbulence,
– d'étudier les résultats obtenus par une simulation LES d'un bâtiment dans un écoulement d'air turbulent.

Langue / Language
Français

Mots-clés anglais / English Keywords
Turbulence, Large-Eddy Simulation Incompressible fluid ; Vorticity ; Large scale ; Modelling ; Turbulence structure.

Localisation / Location
LPBS, Réunion (974)