Synthetic Inflow Condition for Large Eddy Simulation (Synthetic Eddy Method)

EVELYN OTERO

KTH Computer Science and Communication

Master of Science Thesis
Stockholm, Sweden 2009
Synthetic Inflow Condition for Large Eddy Simulation (Synthetic Eddy Method)

EVELYN OTERO

Master’s Thesis in Numerical Analysis (30 ECTS credits) at the Scientific Computing International Master Program Royal Institute of Technology year 2009
Supervisor at CSC was Jesper Oppelstrup
Examiner was Michael Hanke

TRITA-CSC-E 2009:112
ISRN-KTH/CSC/E--09/112--SE
ISSN-1653-5715

Royal Institute of Technology
School of Computer Science and Communication

KTH CSC
SE-100 44 Stockholm, Sweden

URL: www.csc.kth.se
Abstract

Today, whereas aeronautical CFD applications are mainly based on Reynolds Average Navier-Stokes modeling, Large Eddy Simulation (L.E.S.) is becoming more used for some of them. LES needs the instantaneous inlet velocity as turbulent inflow boundary conditions which motivates the implementation and validation of a synthetic turbulent inflow condition for LES. The so called 'Synthetic Eddy Method' (SEM), which is a stochastic algorithm that generates instantaneous velocity, has been studied and validated in homogeneous isotropic turbulence (HIT). A parametric study was conducted in order to extend the SEM to channel flow and comparisons with other boundary conditions such as the periodic LES were carried out.
Referat

Syntetiska virvlar som inflödesvillkor för
Large Eddy - simulering

Large Eddy - Simuleringar (LES) blir allt vanligare som ersättare för Reynolds-Averaged Navier-Stokes beräkningar för extern och intern aerodynamik. LES-modeller behöver ett turbulent inflöde. I detta arbete studeras Synthetic Eddy-metoden, som med stokastisk metod genererar ögonblickliga hastighetsfält. Metoden har implementerats i en industriell strömningslösare (el5A) och validerats på fall med homogen isotrop turbulens och utvidgats till kanalströmning. SEM jämförs med andra metoder som periodisk LES som "återanvänderden strömning som lämnar området."
Acknowledgments

Firstly I would want to thank my supervisor at CERFACS Jean-François Boussuge who followed carefully the progression and quality of the work through a defined schedule and was always available for any assistance. Moreover I wanted to express my grateful to Hugues Deniau, Marc Montagnac and Guillaume Puigt for their regular support regarding the implementation inside the elsA code or technical problems. Then I appreciated the work atmosphere at CERFACS, namely, very professional and international through many seminars, as well as very convivial and friendly. Finally I thank my professors from KTH, Michael Hanke and my supervisor, Jesper Oppelstrup, as well as Ulrich Rüde, from the University of Erlangen, to have allowed me to realize my master’s thesis abroad during the double degree program, being always motivating and available for any advice.
Contents

1 Introduction 1

2 Background 3
  2.1 LES Governing Equations 3
  2.2 Sub-Grid Scale Model 5
    2.2.1 Sub-grid viscosity concept 5
    2.2.2 Smagorinsky model 6
    2.2.3 WALE (Wall Adapting Local Eddy-Viscosity) model 7
  2.3 Numerical Method 8
    2.3.1 Flux computation 9
    2.3.2 Time integration 11
  2.4 Generation of Inflow Boundary Conditions for LES 13
    2.4.1 Introduction 13
    2.4.2 Recycling Methods 13
    2.4.3 Synthetic Turbulence 16

3 The Synthetic Eddy Method 21
  3.1 Principle: SEM basic equations 21
    3.1.1 Definition of the box of eddies 22
    3.1.2 Computation of the velocity signal 23
    3.1.3 Convection of the population of eddies 23
    3.1.4 Regeneration of the eddies out of the box 24
    3.1.5 Synthesis 24
  3.2 Signal characteristics 25
    3.2.1 A stationary ergodic process 25
    3.2.2 Mean flow and Reynolds stresses 25
    3.2.3 Higher order statistics 28

4 Results 31
  4.1 Homogeneous Isotropic Turbulence (HIT) 31
    4.1.1 Definition 31
    4.1.2 Simulation setup and parameters 32
    4.1.3 Statistics 33
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.1.4</td>
<td>Parameters influence: Spatial and temporal analysis</td>
<td>34</td>
</tr>
<tr>
<td>4.1.5</td>
<td>Transition to a Non-HIT</td>
<td>40</td>
</tr>
<tr>
<td>4.2</td>
<td>Channel Flow Computations</td>
<td>41</td>
</tr>
<tr>
<td>4.2.1</td>
<td>Spatially Developing Channel Flow</td>
<td>41</td>
</tr>
<tr>
<td>4.2.2</td>
<td>Periodic Channel Flow</td>
<td>47</td>
</tr>
<tr>
<td>4.2.3</td>
<td>Statistics</td>
<td>48</td>
</tr>
<tr>
<td>5</td>
<td>Discussion</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td>Bibliography</td>
<td>53</td>
</tr>
</tbody>
</table>
Turbulent fluid flow simulation have shown to have a relevant aspect in some industrial applications, for instance in aeronautics. In fact we can observe this phenomena in the flows around aircrafts, in the mixing of fuel and air in the engines. Being the cause of several problems like noise, instability due to turbulent vortices on the surface of the aircraft, an accurate simulation of turbulent flow is extremely necessary for a well operating system, improving design processes for cheaper and faster aircrafts constructions.

For this purpose there are different simulation methods depending on the area considered (level of turbulence) or the kind of application we are interested in. Only for low Reynolds numbers and simple geometries is a Direct Numerical Simulation (DNS) of the Navier-Stokes equations feasible without any turbulence model. For high Reynolds numbers, Reynolds-averaged Navier-Stokes (RANS) simulations offer another alternative, that resolves only the mean motion and relies on a turbulence model to represent the turbulence. Between RANS and DNS lies Large-Eddy Simulation (LES), which resolves large eddies while the smaller ones are modeled. This method is becoming widely used in some industrial applications and this project focuses on LES and specification of realistic inlet boundary conditions which play a major role in the computation. Thus the project will implement and validate a turbulent inflow condition for LES. This condition should be able to represent accurately the turbulence characteristics. We study the 'Synthetic Eddy Method' (SEM), which is a stochastic algorithm that generates instantaneous velocity. Hence, this report will focus on this method, starting from an introduction to the LES theory, followed by the presentation of the Synthetic Eddy Method, and finishing by its implementation and results analysis comparing it to another method as the periodic LES.
Chapter 2

Background

Large-eddy simulations (LESs) of turbulent flows are actually extremely powerful techniques consisting in the elimination of scales smaller than some scale $\Delta x$ by a proper low-pass filtering [26].

LES comes from the meteorological community with the pioneering work of Smagorinsky [43]. The first application in an engineering flow was performed by Deardorff [5], for a fully developed channel flow and since the 1990s the number of applications and flow conditions treated with LES has kept growing. In fact LES has contributed to a blooming industrial development in the aerodynamics of cars, trains, and planes; propulsion, turbo-machinery; thermal hydraulics; acoustics; and combustion as well as in many applications in meteorology.

The Kolmogorov theory defines the small scales in the inertial and dissipative range as having a universal behavior and being parametrizable solely by the energy transfer rate entering the cascade. Large-eddy simulation (LES) will then pose a very difficult theoretical problem of subgrid-scale modeling, namely, how to account for small-scale dynamics in the large-scale motion equations. Hence the aim of using LES is to reduce the considerable computational cost of DNS simulating only in a direct way large scales of turbulence and modeling the smaller scales. For wall flows at high Reynolds numbers, LES and proper computational mesh refinement can save 99% of the effort devoted to resolving scales in the dissipation range (Pope, [38]) compared to DNS of isotropic turbulence at moderate Reynolds number.

The mathematical expressions related will be developed in this section as well as the different models for the small eddy scales. Moreover the numerical method used in the elsA code (cf. chapter 4.2) will be exposed followed by an introduction to the different methods regarding the generation of Inflow Boundary Conditions for LES.

2.1 LES Governing Equations

In this section, based on the Ref. [18], the LES governing equations of the motion of the flows for averaged and filtered quantities are derived and later details of the
turbulence models used to close the equations are provided. Although the software elsA solves the compressible Navier-Stokes equation, we will restrict ourselves to the simulation of incompressible flows where the mass density $\rho$ and the dynamic viscosity $\mu$ are constant. In this case, the Navier-Stokes equations governing the evolution of the velocity $u$ and pressure $p$ of the fluid read

$$\rho \frac{\partial u}{\partial t} + \nabla \cdot (\rho u \otimes u) = -\nabla p + \nabla \cdot \sigma \quad (2.1)$$

$$\nabla \cdot u = 0 \quad (2.2)$$

The viscous stress tensor is $\sigma = 2\mu S$ for a Newtonian fluid where

$$S = \frac{1}{2} (\nabla u + (\nabla u)^T) \quad (2.3)$$

is the strain rate tensor.

However since the direct numerical simulation of the above Navier-Stokes equations requires an extreme computational cost due to the several scales involved, a smoothing operator has to be applied to the exact solution $u$ of the Navier-Stokes equations. The complexity of the system to be solved will thus be reduced.

Assuming that the LES spatial filter and the derivative operators commute, governing equations for the filtered/averaged quantities $\overline{u}$ and $\overline{p}$ can then be derived, where $\overline{u}$ corresponds to the LES averaging operators

$$\rho \frac{\partial \overline{u}}{\partial t} + \nabla \cdot (\rho \overline{u} \otimes \overline{u}) = -\nabla \overline{p} + \nabla \cdot (\overline{\sigma} + \tau) \quad (2.4)$$

$$\nabla \cdot \overline{u} = 0 \quad (2.5)$$

And the subgrid-scale stress tensor is $\tau = -\rho(\overline{u} \otimes \overline{u} - \overline{u} \otimes \overline{u})$

Hence the LES representations of the flow field are seen as two levels of description of the exact solution $u$ of the Navier-Stokes equations.

Labourasse and Sagaut [20] defined the LES spatial filtering operator (see Eq. (2.6)) in the general framework of multilevel methods relying on different scale separation operators.

In fact the separation of large scales and small scales is done via a low-pass filtering operation, which is defined as a convolution product

$$\overline{u} = G \ast u = \int_{-\infty}^{+\infty} u(y) G_{\Delta}(x - y) dy \quad (2.6)$$
where $G_\triangle$ is the convolution kernel characteristic of the filter used and $\triangle$ is the associated cutoff scale.

By the convolution product of the Eq. (2.6), motions smaller than the cutoff scale are removed, thus the velocity $u$ becomes smoother.

As noted before, applying this filter to the Navier-Stokes equations results in a new set of equations for the filtered velocity $u$ (see Eq. (2.4) and (2.5)). It differs from the original Navier-Stokes equations by the presence of the subgrid-scale stresses $\tau = -\rho(\overline{u} \otimes \overline{u} - \overline{u} \otimes \overline{u})$ which needs to be modeled as a function of known averaged quantities in order to close the set of equations.

The most popular model for the subgrid-scale stresses are eddy viscosity models, such as the model proposed by Smagorinsky [43] and the WALE (Wall Adapting Local Eddy-Viscosity) [13] model presented as follows.

### 2.2 Sub-Grid Scale Model

The effect of the small-scale turbulence on the resolved scales is modeled using a subgrid scale (SGS) model. The SGS model commonly employs information from the smallest resolved scales as the basis for modeling the stresses of the unresolved scales.

#### 2.2.1 Sub-grid viscosity concept

An analogy with the kinetic theory of gases has been used to model the direct cascade of energy towards subgrid-scales. We assume that the mechanism of energy transfer of the resolved scales towards subgrid-scales can be represented by a diffusion term using a subgrid viscosity $\mu_{sm}$. A Boussinesq formulation is used where the deviatoric part of the subgrid constraints tensor is linked to the deformation tensor of the resolved field by:

$$
\tau^t_{ij} - \frac{1}{3} \tau^t_{kk} \delta_{ij} = -2\mu_{Sm} T_{ij}
$$

Where $T_{ij}$ is the deviatoric part of the deformation tensor:

$$
T_{ij} = S_{ij} - \frac{1}{3} S_{kk} \delta_{ij},
$$

(2.8)

The subgrid viscosity $\mu_{sm}$ has to be modeled. We consider in this study that $\tau^t_{kk}$ is absorbed by the pressure term. We redefine then a modified pressure $\Theta_P$ and a modified temperature $\Theta_T$:

$$
\Theta_P = P - \frac{1}{3} \tau^t_{kk}
$$

(2.9)
The state equation is written \[10\]:

\[
\Theta_P \approx \rho R \Theta_T
\]

The subgrid viscosity \( \mu_{sm} \) with a scale of length \( l_0 \) and a time scale \( t_0 \) characteristics of the subgrid quantities. The models will have to determine these characteristic scales and here we define two of them, the Smagorinsky and the WALE model which is used for the computations of this project.

### 2.2.2 Smagorinsky model

This is an old model, at the origin of other important models. It is based in the resolved scales and uses a similar approach to the model of Prandtl mixture length. If we assume that the cutoff scale \( \Delta_c \) imposed by the filter is representative of the subgrid modes, then we obtain the length scale \( l_0 \):

\[
l_0 = C_s \Delta_c
\]

A time scale is defined by assumption of local equilibrium between the kinetic energy production rate, the dissipation energy rate by viscosity in internal energy and the kinetic energy flux through the cutoff imposed by the filter:

\[
\frac{1}{t_0} = (2S_{ij}S_{ij})^{1/2}
\]

The eddy viscosity of the subgrid structures \( \left\{ \nu_{sm} \right\} \) is proportional to the product of a length by a velocity, thus it is written

\[
\nu_{sm} = C_s^2 \Delta^2 \sqrt{2S_{ij}S_{ij}}
\]

with \( C_s \) the constant of the model to determine and (here)

\[
S_{ij} = \frac{1}{2} \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right)
\]

If we assume the energetic cascade of Kolmogorov, the constant \( C_s \) can be evaluated (Ref.\[23\]) in a way such that the subgrid dissipation is equivalent to the unresolved scales dissipation:
2.2. SUB-GRID SCALE MODEL

\[ C_s = \frac{1}{\pi} \left( \frac{3C_k}{2} \right)^{-3/4} \approx 0.18 \]  

(2.16)

where \( C_k = 1.4 \) is the Kolmogorov constant. In practice, this model is known to be too dissipative for flows where an average shear is present such as boundary layer. Moreover, it cannot predict dynamics of weakly turbulent flows or transitional flows.

2.2.3 WALE (Wall Adapting Local Eddy-Viscosity) model

The WALE model (Wall Adapting Local Eddy-Viscosity) is based on the Smagorinsky approach where the time scale, characteristic of the subgrid scales, is build with the deformation tensor \( \{S_{ij}\} \) and the rotation tensor \( \{\Omega_{ij}\} \). This new formulation can take into account the turbulent regions where the vorticity is bigger than the deformation rate and it can also provide, without using any damping function, the good behavior of the subgrid viscosity \( \nu_{sm} \) in the proximity to the walls. In fact it has been developed as a solution to the main problems coming from the Smagorinsky model, which are [7]:

- The incapacity to predict the laminar-turbulent transition
- A too important subgrid dissipation (phenomenon changing with the distance to the wall)

Nicoud and Ducros [7] proposed the following subgrid viscosity definition:

\[ \nu_{sm} = C_m \Delta_c^2 \{OP\}(\vec{x}, t) \]  

(2.17)

Where \( C_m \) is the constant of the model, \( \Delta_c \) the characteristic length scale of the subgrid structures and \( OP \) a spatial and temporal operator defined from the resolved field. The Smagorinsky model is obtained for setting \( \{OP\} = (2\{S_{ij}\}\{S_{ij}\})^{1/2} \). However, and in accordance to the recent observations [10], this operator \( OP \) is not only linked to the deformation rate. The WALE model defines \( \nu_{sm} \) as function of the deformation rate and of the rotation rate as:

\[ \nu_{sm} = C_d^2 \Delta_c^2 \frac{(S_{ij}^d S_{ij}^d)^{3/2}}{\{(S_{ij})\{S_{ij}\})^{5/2} + (S_{ij}^d S_{ij}^d)^{5/4}} \]  

(2.18)

with

\[ S_{ij}^d = \frac{1}{2} \left( \left[ \frac{\partial U_i}{\partial x_j} \right]^2 + \left[ \frac{\partial U_j}{\partial x_i} \right]^2 \right) - \frac{1}{3} \left[ \frac{\partial U_k}{\partial x_k} \right]^2 \delta_{ij} \]  

(2.19)
and

\[ C_m^2 \approx 10.6C_s^2 \approx 0.58 \]  \hspace{1cm} (2.20)

\section*{2.3 Numerical Method}

Among the important number of numerical methods simulating the viscous fluid, some of them will be presented on this section taken from the Ref.[6]. But firstly we need to define the hypotheses used for external aerodynamics which are the followings:

- The air is considered as being a perfect gas
- The kinetic viscosity of the air depends only on the temperature. Exemple: the law of Sutherland

\[ \frac{\mu}{\mu_\infty} = \left( \frac{T}{T_\infty} \right)^{3/2} \frac{T_\infty + S_1}{T + S_1} \]  \hspace{1cm} (2.21)

Where \( T \) is the temperature, \( \mu_\infty \) the kinetic viscosity for the reference temperature \( T_\infty \) and \( S_1 \) a constant (110 \( \cdot \) 3\( \times \) for air).

- The conductive heat flux \( \overrightarrow{q} \) is given by the Fourier law

\[ \overrightarrow{q} = -\lambda \overrightarrow{\nabla}T \]  \hspace{1cm} (2.22)

With these hypotheses, the Navier-Stokes equations are written:

\[ \frac{\partial \overrightarrow{W}}{\partial t} + div \overrightarrow{F} = 0 \]  \hspace{1cm} (2.23)

With \( \overrightarrow{W} \) the vector of the conservative variables \( \begin{pmatrix} \rho \\ \rho u \\ \rho v \\ \rho w \\ \rho E \end{pmatrix} \) where \( E \) is the kinetic energy, \( \overrightarrow{F} \) the flux matrice, \( \overrightarrow{F} \) equals to \( \overrightarrow{F} = \overrightarrow{f} - \overrightarrow{f}_v \) where \( \overrightarrow{f} \) is the matrice of the convectiv flux and \( \overrightarrow{f}_v \) is the matrix of the viscous flux.
2.3. NUMERICAL METHOD

\[ f = \begin{pmatrix} \rho u & \rho v & \rho w \\ \rho u v & \rho v^2 + p & \rho p w \\ \rho u w & \rho p w & \rho w^2 + p \\ u(\rho E + p) & v(\rho E + p) & w(\rho E + p) \end{pmatrix} \]

\[ f_v = \begin{pmatrix} 0 & 0 & 0 \\ \tau_{xx} & \tau_{xy} & \tau_{xz} \\ \tau_{xy} & \tau_{yy} & \tau_{yz} \\ (\tau u)_x - q_x & (\tau u)_y - q_y & (\tau u)_z - q_z \end{pmatrix} \]

with \( \overline{\tau} = \mu \left( \frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - \frac{2}{3} \mu \frac{\partial U_k}{\partial x_k} \delta_{ij} = 2 \mu \overline{\tau} - \frac{2}{3} \mu \text{div}(\overline{U}) \) the shear constraints tensor and \( \overline{q} \) the vector of the conductive heat flux computed with the Fourier law.

In the context of a finite volume approach, we integrate the Eq.(2.23) over a control volume \( \Omega \). The problem to solve becomes:

\[ \int_\Omega \frac{\partial \overline{W}}{\partial t} dV + \oint_{\partial \Omega} \overline{F}.\overrightarrow{n}dS = 0 \quad (2.24) \]

Where \( \overrightarrow{n} \) is the normal vector to the surface of the control volume.

In the code of computation of elsA, it has been decided to decouple the time integration and the spatial discretization, therefore we can choose the time integration scheme independently of the spatial discretization method.

In order to solve numerically these non-stationary flow problems, we have to treat the following points:

1. How to compute the flux (computation of \( \oint_{\partial \Omega} \overline{F}.\overrightarrow{n}dS \)) which corresponds to the spatial discretization
2. The type of time integration
3. The linearization (if it’s necessary: in the implicit case for example)
4. A method of linear system resolution

In this section the two first points will be treated, and more details can be found on the Ref. [6].

2.3.1 Flux computation

2.3.1.1 Convective flux computation

This section presents two methods of computation of convective flux. The term \( \oint_{\partial \Omega} \overline{F}.\overrightarrow{n}dS \) of the Eq.(2.24) can be decomposed as \( \oint_{\partial \Omega} \overline{F}.\overrightarrow{n}dS - \oint_{\partial \Omega} \overline{F}.\overrightarrow{v}dS \) but we will focus on the computation of the first term.
Scheme of Jameson with artificial dissipation

Since this scheme is based on a central approximation of flux, the resultant numerical scheme is stabilized by artificial dissipation. Moreover, structured grid will be considered and will be presented in 2D as following:

The term $\int_{\partial \Omega} f \cdot \vec{n} dS$ on the cell $(i, j)$ can be then decomposed into: $f_{(i+1/2,j)} \bar{\vec{f}} \vec{n} dS + f_{(i-1/2,j)} \bar{\vec{f}} \vec{n} dS + f_{(i,j+1/2)} \bar{\vec{f}} \vec{n} dS + f_{(i,j-1/2)} \bar{\vec{f}} \vec{n} dS$

The numerical flows $\vec{F}_{i+1/2,j} = f_{(i+1/2,j)} \bar{\vec{f}} \vec{n} dS$ for the central scheme are written:

$$\vec{F}_{i+1/2} = \bar{\vec{f}}_{i+1/2} \tilde{s}_{i+1/2} - \tilde{d}_{i+1/2}$$  \hspace{1cm} (2.25)

with $\bar{\vec{f}}_{i+1/2}$ the matrix of the convective flux computed on the interface $(i+1/2, j)$ of the control volume $(i, j)$, $\tilde{s}_{i+1/2}$ the surface vector (equal to the normal vector of the interface multiplied by its surface) and $\tilde{d}_{i+1/2}$ a dissipation term.

Roe scheme

The numerical flux for the scheme of Roe is:

$$\vec{F}_{i+1/2} = \frac{1}{2} \left[ \bar{\vec{f}}(\bar{W}^L_{i+1/2}) + \bar{\vec{f}}(\bar{W}^R_{i+1/2}) \right] \cdot \tilde{s}_{i+1/2}$$

$$- \frac{1}{2} \left| \bar{A}(\bar{W}^L_{i+1/2}, \bar{W}^R_{i+1/2}) \right| (\bar{W}^R_{i+1/2} - \bar{W}^L_{i+1/2})$$  \hspace{1cm} (2.26)

Where $\bar{A}$ is the matrix of Roe and $\bar{W}^L_{i+1/2}$ and $\bar{W}^R_{i+1/2}$ are the values of the conservative variables on the interfaces. These values are interpolated by noncentral schemes of order more or less big depending on the desired precision.
2.3. NUMERICAL METHOD

2.3.1.2 Computation of the Diffusive flux

The computation of the diffusive flux $F$ is always done in elsA \cite{8}(para.3.5.2) with a central approach.

2.3.1.3 Conclusion

Thus after the application of the spatial discretization following the previous methods, the Eq.(2.24) becomes:

$$\frac{d}{dt} \left( V_{(i,j)} \overrightarrow{W}_{(i,j)} \right) + \overrightarrow{R}_{(i,j)} = 0$$ \hspace{1cm} (2.27)

With $V_{(i,j)}$ the volume of the cell, and

$$\overrightarrow{R}_{(i,j)} = \overrightarrow{F}_{(i+1/2,j)} - \overrightarrow{F}_{(i-1/2,j)} + \overrightarrow{F}_{(i,j+1/2)} - \overrightarrow{F}_{(i,j-1/2)}$$

$$\overrightarrow{F}_{v(i+1/2,j)} - \overrightarrow{F}_{v(i-1/2,j)} + \overrightarrow{F}_{v(i,j+1/2)} - \overrightarrow{F}_{v(i,j-1/2)}$$ \hspace{1cm} (2.28)

Which corresponds to the set of convective and diffusive flux over the control volume.

2.3.2 Time integration

2.3.2.1 Explicit method : Runge Kutta exemple

In order to integrate in time the Eq.(2.27), a Runge Kutta scheme can be applied as the following, for a 4 steps case:

$$\overrightarrow{W}^{(1)} = \overrightarrow{W}^n - \alpha_1 \frac{\Delta t}{V} \overrightarrow{R}^{(n)}$$
$$\overrightarrow{W}^{(2)} = \overrightarrow{W}^n - \alpha_2 \frac{\Delta t}{V} \overrightarrow{R}^{(1)}$$
$$\overrightarrow{W}^{(3)} = \overrightarrow{W}^n - \alpha_3 \frac{\Delta t}{V} \overrightarrow{R}^{(2)}$$
$$\overrightarrow{W}^{(4)} = \overrightarrow{W}^n - \alpha_4 \frac{\Delta t}{V} \overrightarrow{R}^{(3)}$$
$$\overrightarrow{W}^{n+1} = U^{(4)}$$

Explicit methods have conditional stability. In fact using this time integration and a second order central scheme for the spatial discretization for the model equation $\frac{\partial u}{\partial t} + a \frac{\partial u}{\partial x} = 0$, the theoretic criteria of linear stability requires a Courant-Friedrichs-Lewy number as $CFL = \frac{a \Delta t}{\Delta x} \leq 2\sqrt{2}$ (cf.\cite{14}) but a stricter limit exits for the boundary layer because $\Delta x$ becomes very small. This corresponds to the viscous time step limit : $\Delta t \leq C.\Delta x^2$. For non-stationary simulations of viscous fluid, the size of the cells can become very small which means that the limitation in the integration step imposed by the CFL condition can sometimes require an important computational cost or can even cause an impossible simulation.
2.3.2.2 Implicit method

For the time integration, we use multi-step linear schemes. If we use a one step scheme (Backward Euler scheme), the Eq.(2.27) becomes:

\[ V \frac{\overrightarrow{W}^{n+1} - \overrightarrow{W}^n}{\Delta t} + \overrightarrow{R}(\overrightarrow{W}^{n+1}) = 0 \]  \hfill (2.29)

And the linearization of the residual at the time n+1 gives

\[ \overrightarrow{R}^{n+1} = \overrightarrow{R}^n + \frac{\partial \overrightarrow{R}}{\partial \overrightarrow{W}} |^{\Delta \overrightarrow{W}}^{n} + \theta(\Delta t^2) \]  \hfill (2.30)

Thus the Eq.(2.29) becomes:

\[ \left( \frac{V}{\Delta t} \overrightarrow{W} + \frac{\partial \overrightarrow{R}}{\partial \overrightarrow{W}} \right)^n \Delta \overrightarrow{W}^n = -R(\overrightarrow{W}^n) \]  \hfill (2.31)

which can be more linearized (cf. [6])

2.3.2.3 Dual time step or Dual time Stepping (DTS)

DTS is another temporal integration method, which consist in using a fictive time step solving an equivalent problem pseudo-stationary. The non stationary problem at the time \((n + 1)dt\) leads to the resolution of the following equation:

\[ \overrightarrow{R}^* (\overrightarrow{W}^{n+1}) = \left( \frac{dV}{dt} \right)^{n+1} + \overrightarrow{R}(\overrightarrow{W}^{n+1}) = 0 \]  \hfill (2.32)

Three advantages concern this method: Firstly the approximation of the temporal derivative can be realized for any order (cf. [3]). The precision in time can be thus improved in comparison with the Backward Euler scheme of order 1. The second advantage is that all the methods of convergence acceleration can be used since the problem to solve is a pseudo-stationary system at each time step. We can, for example, use the methods employed today in stationary.

Finally, this method allows to get rid of the constraint regarding the stability for explicit simulations. Using the dual time step method, the time step can be determined in function of the characteristic time of the phenomena to be observed.
2.4 Generation of Inflow Boundary Conditions for LES

This section mainly summarizes the presentation of Jarrin [18](Chap. 3) regarding the generation of inflow boundary conditions for LES and DNS and also refers to Ref. [28].

2.4.1 Introduction

Once we have described an appropriate turbulence model it’s of great importance the specification of realistic inlet boundary conditions which play a major role in the accuracy of a numerical simulation. Whereas for RANS approaches, only mean profiles for the velocity and the turbulence variables are required, for large-eddy and direct-numerical simulations, turbulent unsteady inflow conditions have to be prescribed which makes the generation of inflow data an important issue. Actually DNS or LES results show, particularly in the cases of a plane jet [25], how sensitive a spatially developing boundary layer [45] or a backward facing step [31] are to inflow conditions. Ideally the simulation of the upstream flow entering the computational domain would provide realistic inlet conditions to the main simulation. For obvious reasons however, the computational domain cannot be extended upstream indefinitely, and so approximate turbulent inlet conditions must be specified. Exceptions include LES or DNS of transition processes, where the inlet boundary is located in a region where the flow is laminar [39], [41]. In this case, random disturbances are superimposed on a laminar profile to trigger the transition process, and no turbulent fluctuations are required at the inlet. However, this method cannot be extended to inlet boundaries where the flow is turbulent, since simulating the whole transition process would be far too costly. In order to limit the computational cost of LES or DNS of spatially evolving flows, the boundaries have to be placed as close as possible to the region of interest. This in turn requires the approximate boundary conditions to be as accurate as possible in order to limit their effect on the flow inside the domain. Therefore, the length of the transition region (where the approximate inflow data generated at the inlet of the LES or DNS domain develops towards a more physical state) must be made as short as possible. A very effective way to avoid this problem is to use periodic boundary conditions, but this technique is restricted to a few simple geometries and test cases. For more general cases different techniques exist which are going to be discussed as following in order to choose the most appropriate one.

2.4.2 Recycling Methods

Today these methods are the most accurate in the specification of turbulent fluctuations for LES or DNS. They consist in running a precursor simulation whose only role is to provide the main simulation with accurate boundary conditions.
2.4.2.1 Periodic boundary conditions

Presentation

Periodic boundary conditions in the mean flow direction can be applied to the precursor simulation if the turbulence at the inlet of the main simulation can be considered as fully developed (which is often the case for internal flows such as ducts, channels or pipes). This means that the flow at the outflow plane will be recycled and reintroduced at the inlet so that the simulation generates its own inflow data. In fact Fig. 2.2 shows the overall phenomenon: at each time step instantaneous velocity fluctuations in a plane at a fixed streamwise location are extracted from the precursor simulation and prescribed at the inlet of the main simulation.

![Figure 2.2. Sketch of a LES using a precursor simulation dedicated to the generation of the inflow data. Taken from Ref. [18]](image)

Special care has to be taken to initialize the flow field properly so that turbulence can be generated as the simulation evolves. In general, the flow is initialized with a mean velocity profile plus a few unstable Fourier modes [16]. However the application of this method is limited to simple fully developed flows since, as already mentioned, periodic boundary conditions can only be used to generate inflow conditions which are homogeneous in the streamwise direction.

Applications

Several scientists adopted this approach as Kaltenbach [15] and Friedrich and Arnal [11] who extracted velocity planes from a precursor periodic channel flow to generate inflow data for a LES of a plane diffuser and a LES of a backward-facing step respectively.
2.4. GENERATION OF INFLOW BOUNDARY CONDITIONS FOR LES

2.4.2.2 Rescaling/recycling technique

Presentation

A more flexible technique for a zero pressure gradient spatially developing boundary layer is proposed by Lund [45]. A sketch of the computational set-up is shown in Fig. 2.3.

The procedure was shown by Lund [45] to result in a spatially evolving boundary layer simulation that generates its own inflow data. Planes of velocity data can then be saved from a precursor simulation using this procedure and used as inflow conditions for the main simulation. Actually the method works as following: the velocity at the rescaling station, which means in a plane located several boundary-layer thicknesses downstream of the inlet, is used to evaluate the velocity signal at the inlet plane. Moreover this velocity field at the rescaling station is decomposed into its mean and fluctuating part and the scaling is applied to the mean and the fluctuating parts in the inner and outer layers separately, to account for the different similarity laws that are observed in these two regions. The rescaled velocity is finally reintroduced as a boundary condition at the inlet.

Applications and Evolution

Aider and Danet [1] used this method to generate inlet conditions for turbulent flow over a backward-facing step and Wang and Moin [46] to generate inlet conditions for a hydrofoil upstream of the trailing edge, in a region where the pressure gradient was minimal.
CHAPTER 2. BACKGROUND

Subsequently Sagaut [34] proposed an extension of the original method of Lund [45] to compressible turbulence using rescaling and recycling of the pressure and temperature fluctuations in addition to the usual operations performed in the original method.

Moreover a more robust variant of the original method of Lund was proposed by Ferrante and Elghobashi [9] who noticed that Lund’s method was very sensitive to the initialization of the flow field. In fact they were unable to generate fully developed turbulence from the initialization procedure recommended in Lund [45] (mean velocity profile superimposed with random uncorrelated fluctuations). In their new approach the flow field was initialized using synthetic turbulence with a prescribed energy spectrum and shear stress profile.

Additionally, another modification was recently proposed by Spalart [35] to the original method of Lund [45] regarding the reduction of the computational cost of generating the inflow data. The recycling station should be taken much closer to the inflow and the precursor simulation eliminated, carrying out in the main simulation domain the recycling procedure so that it generates its own inflow data. Li [32] proposed also another procedure to reduce the storage requirement and the computational cost of running a precursor simulation prior to the main simulation.

2.4.3 Synthetic Turbulence

These methods synthetize inflow conditions using some kind of stochastic procedure and don’t need precursor simulation or rescaling of a database created from a precursor simulation. These stochastic procedures are referred to as synthetic turbulence generation methods and use random number generators to construct a random velocity signal which resembles turbulence. However, the synthesized turbulence represents only a crude approximation of real turbulence. In fact the underlying assumption on which they are based, is that a turbulent flow can be approximated by reproducing a set of low order statistics such as mean velocity, turbulent kinetic energy, Reynolds stresses, two-point and two-time correlations. Higher order statistics such as the terms in the budget of the Reynolds stresses (the rate of dissipation, the turbulent transport or the pressure-strain term) are not usually reproduced. Additionally, the synthesized turbulence may show different structures from the one corresponding to a real flow. It could also undergo a transition process before it developing towards a more physical state if the structure of the turbulent eddies and their dynamics is not accurately reproduced. In recent years several synthetic turbulence generation methods have been proposed which are going to be discussed as following differentiating the ones working in physical space (referred to as algebraic methods) from the methods working in Fourier space (referred to as spectral methods).
2.4. GENERATION OF INFLOW BOUNDARY CONDITIONS FOR LES

2.4.3.1 Algebraic methods

Algebraic methods work with sets of random numbers and by performing operations on them, try to fit the target statistics of turbulence. The synthetic fluctuations can be generated from a set of independent random numbers \( r_i \) taken from a normal distribution \( \mathcal{N}(0, 1) \) of mean \( \mu = 0 \) and variance \( \sigma = 1 \) and rescaling them such that the fluctuations have the correct turbulent kinetic energy \( k \). These computed fluctuations are then added on to a mean velocity profile \( U \). The inflow signal corresponds to the following expression:

\[
 u_i = U_i + r_i \sqrt{\frac{2}{3}} k \tag{2.33}
\]

where the \( r_i \) are taken from independent random variables for each velocity component at each point and at each time step. The procedure below generates a random signal which reproduces the target mean velocity and kinetic energy profiles. However all cross correlations between the velocity components and the two-point and two-time correlations are zero. However this method can be improved by correlating the components of the velocity, which was proposed in Appendix B of Lund [45]. Actually the Cholesky decomposition \( A \) of the Reynolds stress tensor \( R_{ij} \) can be used to reconstruct a signal which matches the target Reynolds stress tensor, but obviously this requires a first knowledge of the stress tensor data.

\[
 u_i = U_i + r_j a_{ij} \tag{2.34}
\]

\[
 R = AA^T \tag{2.35}
\]

This procedure allows the basic random procedure to reproduce the target cross-correlations \( R_{ij} \) between velocity components \( i \) and \( j \) if the random data satisfy the necessary conditions \( \langle r_i r_j \rangle = \delta_{ij} \) and \( \langle r_i \rangle = 0 \). This is the case when the \( r_i \) are independent random numbers taken from a normal distribution \( \mathcal{N}(0, 1) \). In the following of the thesis, the procedure described above in Eq. (2.34) and Eq. (2.35) will be referred to as the random method. In real turbulence, the cascade of energy from large scales to small scales is initiated in the large scales, which contain most of the energy; however the random methods presented above still do not yield any correlations either in space or in time. They have energy uniformly spread over all wave numbers which implies an excess of energy in the small scales which dissipate very quickly.

In order to correct the lack of large-scale dominance in the inflow data generated by the random method, Klein [25] discovered a digital filtering procedure described as following. In one dimension the velocity signal \( u'(j) \) at point \( j \) is defined by a convolution (or a digital linear non-recursive filter) as
\[ u'(j) = \sum_{k=-N}^{N} b_k r(j + k) \] (2.36)

with \( b_k \), the filter coefficients, \( N \), connected to the support of the filter and \( r(j + k) \), the random number generated at point \((j + k)\) with a normal distribution \( \mathcal{N}(0, 1) \). Thus the two-point correlations between points \( j \) and \((j + m)\) depend on the filter choice and read

\[ < u'(j)u'(j + m) > = \sum_{k=-N+m}^{N} b_k b_{k-m} \] (2.37)

An extension of this procedure is realized to the time-dependent computation of synthetic velocity field on a plane \((Oyz)\) by generating for each velocity component \( m \), a three-dimensional random field \( r_m(i, j, k) \). The indices \( i, j, \) and \( k \) correspond to the time \( t \), the direction \( y \) and the direction \( z \) respectively. A three-dimensional filter \( b_{ijk} \) is obtained by the convolution of three one-dimensional filters \( b_{ijk} = b_i b_j \ldots b_k \). This is used to filter the random data \( r_m(i, j, k) \) in the three directions \( t, y \) and \( z \),

\[ u'_m(j, k) = \sum_{i'=-N_x}^{N_x} \sum_{j'=-N_y}^{N_y} \sum_{k'=-N_z}^{N_z} b_{i'j'k'} r_m(i', j + j', k + k') \] (2.38)

The random numbers are convected at each time step through the inlet plane using Taylor’s frozen turbulence hypothesis \( r_{i,j,k} \rightarrow r_{i+1,j,k} \) and new random numbers are generated on the plane \( i = 1 \). At the next time step, the new random field is filtered and so on. If the objective would be only to produce homogeneous turbulence, the procedure would thus stop at this moment. However this is generally not the case, for this reason the signal is reconstructed at each time step following Eq. (2.34) to generate target mean velocity and Reynolds stresses profiles. The fluctuations generated have to reproduce exactly the target two-point correlations \( < u'(j)u'(j + m) > \) therefore the filter coefficients \( b_k \) should be computed by inverting Eq. (2.37). Klein [25] assumed a Gaussian shape depending on one single parameter, the lengthscale \( L \), since the two-point autocorrelation tensor is rarely available. Then without inverting Eq. (2.37), the coefficients can be computed analytically. Hence Klein [25] could test with this procedure the influence of the length and time scales (uniform over the inlet plane) on the development of a plane jet and the primary break-up of a liquid jet.

Additionally to the above presented methods which play in fact an important role in this project, other techniques are introduced briefly as following.
2.4. GENERATION OF INFLOW BOUNDARY CONDITIONS FOR LES

2.4.3.2 Spectral methods

Spectral methods use a decomposition of the signal into Fourier modes. Kraichnan [19] was the first to work with a Fourier decomposition to generate a synthetic flow field. The flow domain was initialized with a three-dimensional homogeneous and isotropic synthetic velocity field to study the diffusion of a passive scalar. In fact this method has been used extensively to initialize velocity fields in the study of the temporal decay of homogeneous isotropic turbulence (Rogallo, [40]). Since the signal is homogeneous in the three directions of space, it can be decomposed in Fourier space, where $k$ is a three-dimensional wavenumber

$$u'(x) = \sum_k \hat{u}_k e^{-ik\cdot x}$$

and finally we obtain the following synthesized velocity field,

$$u'(x) = \sum_k \sqrt{E(|k|)} e^{-i(k\cdot x + \Theta_k)}$$

Where each complex Fourier coefficient $\hat{u}_k$ is given a defined amplitude calculated from a prescribed isotropic three-dimensional energy spectrum $E(|k|)$ and a random phase $\Theta_k$, taken from a uniform distribution on the interval $[0, 2\pi]$. However Lee [42] proposed an adaptation of this method for generating inlet boundary conditions for the simulation of spatially evolving turbulent flows. Supposing that the flow is evolving in the $x$ direction, the signal prescribed at the inlet is thus homogeneous in the transverse directions $y$ and $z$ and stationary in time. Thus this time the signal can be decomposed into Fourier modes as in Eq. (2.39), with the Fourier coefficients calculated from an energy spectrum prescribed in terms of frequency and two transverse wave numbers, and where the phases $\phi$ depend on time in order to avoid generating a periodic signal

$$\hat{u}(k_y, k_z, \omega, t) = \sqrt{E(k_y, k_z, \omega)} \exp(i\phi(k_y, k_z, \omega, t))$$

Based on this spectral method of Lee [42], other procedures have been developed as the one of Le [14] for the generation of inlet conditions for the turbulent boundary layer upstream of a backward facing step. Modifications of the original method of Le [14] requiring only statistical information have been proposed in order to be able to synthetize non-homogeneous turbulence in a general industrial framework.
2.4.3.3 Mixed methods

Until now two classifications have been exposed which however don’t reflect the wide range of methods proposed in the literature. Actually some of them are not just restricted to Fourier or physical spaces but perform operations in both at the same time and for this reason they are called mixed methods. Since these are not of major interest for this project, some of them would be just briefly presented. Firstly Davidson [4] proposed a method to generate inlet conditions for LES or DNS based on both Fourier decomposition and digital filters. In order to create correlations in time, the independent isotropic fluctuations synthesized at each time step \( m \) are filtered in time using an asymmetric time filter,

\[
\hat{u}_i(x)^m = au_i(x)^{(m-1)} + b\hat{u}_i(x)^{(m)}
\]  

where \( a \) and \( b \) are the filter coefficients. The generated fluctuations are then superimposed on a target mean velocity profile. This method has the advantage over the method of Le [14] that the Fourier transform is only performed in two dimensions and that no randomization of the phase angles is necessary to break the periodicity of the signal. As in the method of Le [14] however, the synthesized turbulence is homogeneous. Another mixed formulation comes from the area of structural analysis, where it is often necessary to simulate physical loadings due to atmospheric turbulence, ocean waves or earthquakes.

After having presented the different types of inflow boundary conditions generation, we will focus in an example of algebraic method of synthetic turbulence, namely, the Synthetic Eddy Method.
Chapter 3

The Synthetic Eddy Method

Among the variety of methods available to generate inflow boundary conditions for LES, synthetic turbulence generation methods seem to be a good candidate to address industrial applications of LES:

- The method should require only a small fraction of the overall computational cost,
- It should work for any type of inlet mesh, geometry and flow without requiring any particular intervention from the user,
- The inflow information needed for the method should be kept as simple as possible (mean flow, turbulence intensity, RANS statistics, etc.),

Moreover the approach is grounded on the presence of large scale coherent structures in turbulent flows which carry most of the Reynolds stresses. Since in LES these large scale eddies are resolved, the synthetic velocity signal specified at the inflow should, ideally, represent the contribution of these eddies. Thus the interest in developing this method for Inflow boundary conditions in Large Eddy Simulation.

In the following sections, we describe the construction of a stochastic velocity signal using the Synthetic Eddy Method (SEM) before deriving some exact results concerning the statistical properties of the synthetized signal.

The main idea of this project will then consist in the construction of a synthetic velocity signal which can be written as a sum over a finite number of eddies with random intensities and positions.

3.1 Principle: SEM basic equations

The SEM method is actually based in a set of equations which firstly requires the following input data definition:

- A finite set $S \subset R^3$ of points $S = \{x_1, x_2, \ldots, x_s\}$ on which we want to generate synthetic velocity fluctuations with the SEM.
CHAPTER 3. THE SYNTHETIC EDDY METHOD

And for this set of points considered:

- The mean velocity $U$,
- The Reynolds stresses $R_{ij}$
- a characteristic length scale of the flow $\sigma$

Once this information has been provided, we can start the algorithm composed in the following steps.

3.1.1 Definition of the box of eddies

Firstly it is required to define a box of eddies containing the synthetic eddies as seen in the Fig. 3.1.

![Figure 3.1. Box of eddies B surrounding the set of points $S = \{x_1, x_2, \ldots, x_s\}$ on which the SEM signal is going to be computed.](image)

Actually in this figure we can see the plane of entry where the synthetic velocities will be computed for each grid point and surrounded by a set of eddies inside the box $B$ defined as

$$B = \{(x_1, x_2, x_3) \in \mathbb{R}^3 : x_{i,\text{min}} < x_i < x_{i,\text{max}}, i = \{1, 2, 3\}\} \quad (3.1)$$

where
3.1. PRINCIPLE: SEM BASIC EQUATIONS

\[ x_{i,min} = \min_{x \in S} (x_i - \sigma(x)) \quad \text{and} \quad x_{i,max} = \max_{x \in S} (x_i + \sigma(x)) \]  

(3.2)

3.1.2 Computation of the velocity signal

The velocity fluctuations generated by \( N \) eddies are

\[ u_i(x) = U_i + \frac{1}{\sqrt{N}} \sum_{k=1}^{N} a_{ij} \varepsilon_j f(\sigma(x)) (x - x^k) \]  

(3.3)

where the \( x^k = (x^k, y^k, z^k) \) are the locations of the \( N \) eddies, the \( \varepsilon^k_j \) are their respective intensities and \( a_{ij} \) is the Cholesky decomposition of the Reynolds stress tensor:

\[ R = A A^T \]  

(3.4)

Moreover \( f_{\sigma}(x)(x - x^k) \) is the velocity distribution of the eddy located at \( x^k \).

We assume that the distributions depend only on the length scale \( \sigma \) and define \( f_{\sigma} \) by

\[ f_{\sigma}(x - x_k) = \sqrt{V_B} \sigma^{-3} f\left( \frac{x - x_k}{\sigma} \right) f\left( \frac{y - y_k}{\sigma} \right) f\left( \frac{z - z_k}{\sigma} \right) \]  

(3.5)

where the shape function \( f \) is common to all eddies. \( f \) has compact support \([-\sigma, \sigma]\) and has the normalization \( \|f\|_2 = 1 \).

The position of the eddies \( x^k \) before the first time step are independent from each other and taken from a uniform distribution \( U(B) \) over the box of eddies \( B \) and \( \varepsilon^k_j \) are independent random variables taken from any distribution with zero mean and unit variance. In all simulations carried out in this paper we choose \( \varepsilon^k_j \in \{-1, 1\} \) with equal probability to take one value or the other.

3.1.3 Convection of the population of eddies

The eddies are convected through the box of eddies \( B \) with a constant velocity \( U_c \) characteristic of the flow. In our case it is straight forward to compute \( U_c \) as the averaged mean velocity over the set of points \( S \). At each iteration, the new position of eddy \( k \) is given by

\[ x^k(t + dt) = x^k(t) + U_c dt. \]  

(3.6)
where \( dt \) is the time step of the simulation.

**Remark (Ref. [18]):**

"In order to have a better physical model for our synthetic turbulence, we could have used a local convective velocity \( U_c \) for each eddy (computed at its center for instance) instead of one constant velocity for all eddies. However this would mean some regions are populated with faster moving eddies than others [...]. In all our derivations we required that the distribution of eddies for all times remain uniform over \( B \) which is ensured by taking a constant convective \( U_c \) velocity for all eddies."

### 3.1.4 Regeneration of the eddies out of the box

If an eddy \( k \) is convected out of the box through face \( F \) of \( B \), then it is immediately regenerated randomly on the inlet face of \( B \) facing \( F \) with a new independent random intensity vector \( \varepsilon_j^k \) still taken from the same distribution.

### 3.1.5 Synthesis

The method is based on the following properties (Fig. 3.2 [29]):

- A turbulent flow is a superposition of coherent eddies
- Each eddy is described by a shape function \( f(x) \) with compact support
- Random eddies convecting through \( B \) contribute to the generation of eddies on the inlet plane

![Figure 3.2. Shape functions represented on the inlet plane](image)

And the procedure can be summarized as:

1. Determination of the input data
3.2. SIGNAL CHARACTERISTICS

2. Definition of the box containing the inlet plane and the eddies generated

3. Generation for each eddy $k$ of two random vectors $x_k$ and $\varepsilon_k$ for its location inside the box and its intensity, respectively

4. Computation of the velocities applying Eq. (3.3) for each point on the grid of the inlet plane and corresponding to the set of points $S$.

5. Convection of the eddies through $B$ with velocity $U_c$.

6. Regeneration of the data regarding each eddy $k$ out of the box after displacement: new locations $x_k$ and intensities $\varepsilon_k$

3.2 Signal characteristics

Once the algorithm and the equations have been exposed it’s important to show how the method respects the physics relation. But before we will define the signal generated.(see Ref [18])

3.2.1 A stationary ergodic process

We are going to start proving that it is a stationary random process. Firstly the synthesized signal $u(x, t)$ can be seen as a random process of time, at any location $x$. Moreover it is a function of the random variables $\varepsilon_k$ and $x_k$: At any given time $t$, the position of each eddy $k$ follows a uniform distribution over $B$ and its intensity is either 1 or -1 with equal probability. Secondly the synthesized signal is a stationary process since these random variables $\varepsilon_k$ and $x_k$ also keep the same probability density function for all times, remaining identically distributed during the simulation. A stationary random process which becomes independent of its previous state after a certain time is naturally ergodic (Lumley [24]). Actually the synthetic signal itself becomes independent of its previous states after a certain time has passed (namely after all eddies have been recycled at least once). In fact the recycling process ensures that the position and intensity of each eddy before and after the recycling procedure are independent.

The signal generated by the SEM is thus a stationary ergodic random process.

3.2.2 Mean flow and Reynolds stresses

The statistical properties of the synthesized signal are now going to be studied. We start by studying the mean value of the velocity signal given by Eq. (3.3). By linearity of the statistical mean, we obtain

$$ < u_i > = U_i + \frac{1}{\sqrt{N}} \sum_{k=1}^{N} \left( a_{ij} e_j^k f_{\sigma(x)}(x - x_k^i) \right) $$

(3.7)
CHAPTER 3. THE SYNTHETIC EDDY METHOD

The random variables $x_j^k$ and $e_j^k$ involved in the mean $\langle a_{ij}e_j^k f_{\sigma(x)}(x - x^k) \rangle$ are independent thus

$$\langle a_{ij}e_j^k f_{\sigma(x)}(x - x^k) \rangle = \langle a_{ij}e_j^k \rangle f_{\sigma(x)}(x - x^k)$$

(3.8)

The term $\langle a_{ij}e_j^k \rangle$ simplifies further to $\langle a_{ij}e_j^k \rangle = a_{ij} \langle e_j^k \rangle = 0$ since the intensities of the eddies is either 1 or -1 with equal probability. Then we substitute these relations into Eq. (3.7), the mean of the velocity signal $u_i$ is simply the input mean velocity $U_i$,

$$< u_i > = U_i \quad (3.9)$$

and the fluctuations $u_i'$ around the mean velocity are

$$u_i' = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} a_{ij}e_j^k f_{\sigma(x)}(x - x^k) \quad (3.10)$$

The Reynolds stresses $< u_i'u_j' >$ of the synthesized signal is then calculated. Using the above expression and the linearity of the statistical mean, we obtain

$$< u_i'u_j' > = \frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{N} a_{im}a_{jn} < e_j^k e_l^m f_{\sigma(x)}(x - x^k)f_{\sigma(x)}(x - x^l) >$$

(3.11)

Using again the independence between the positions $x_j^k$ and the intensities $e_j^k$ of the eddies, Eq. (3.11) becomes

$$< u_i'u_j' > = \frac{1}{N} \sum_{k=1}^{N} \sum_{l=1}^{N} a_{im}a_{jn} < e_j^k e_l^m > f_{\sigma(x)}(x - x^k)f_{\sigma(x)}(x - x^l) >$$

(3.12)

If $k \neq l$ or $m \neq n$ the random variables $e_j^k$ and $e_l^m$ are independent and hence $< e_j^k e_l^m > = < e_j^k > < e_l^m > = 0$. If $k = l$ and $m = n$, then $< e_j^k e_l^m > = < (e_j^k)^2 > = 1$ by definition of the intensities of the eddies. Hence we can write,

$$< e_j^k e_l^m > = \delta_{kl}\delta_{mn} \quad (3.13)$$
3.2. SIGNAL CHARACTERISTICS

Using the above result, Eq. (3.11) simplifies to

\[
\langle u_i' u_j' \rangle = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} a_{im} a_{jn} \left\langle \frac{f_{\sigma}^2(x - x^k)}{I_i} \right\rangle
\] (3.14)

The Probability Density Function (PDF) of \( x^k \) is needed in order to compute the term \( I_1 \). \( x^k \) follows a uniform distribution over \( B \) hence by definition its probability density function writes

\[
p_1(x) = \begin{cases} \frac{1}{V_B}, & \text{if } x \in B \\ 0, & \text{otherwise} \end{cases}
\] (3.15)

Thus

\[
I_1 = \int_{R^3} p(y) f_{\sigma}^2(x - y) dy = \frac{1}{V_B} \int_{B} f_{\sigma}^2(x - y) dy
\] (3.16)

Besides by definition of \( B \),

\[
x \in S, \quad y \notin B \quad \implies \quad (x - y) \notin \text{supp}(f_{\sigma})
\] (3.17)

The integral over \( B \) on Eq. (3.16) can then be replaced by an integral over \( R^3 \). Using the definition of \( f_{\sigma} \) and the normalization of \( f \), \( I_1 \) rewrites

\[
I_1 = \frac{1}{V_B} \int_{R^3} f_{\sigma}^2(x - y) dy = 1
\] (3.18)

Finally using the above result into Eq. (3.14) the cross correlation tensor writes

\[
\langle u_i' u_j' \rangle = a_{im} a_{jn} = R_{ij}
\] (3.19)

since \( a_{ij} \) is the Cholesky decomposition of \( R_{ij} \). Hence the Reynolds stresses of the velocity fluctuations generated by the SEM reproduce exactly the input Reynolds stresses \( R_{ij} \).
CHAPTER 3. THE SYNTHETIC EDDY METHOD

3.2.3 Higher order statistics

Moreover in order to understand better the behavior of the signal, higher order moments as the skewness and the flatness, will be computed which will allow us to check afterwards the validity of the SEM implementation.

3.2.3.1 Skewness

To express the final equations of these statistics we have to start with another formulation of the SEM velocity fluctuations by rearranging Eq. (3.3) as

$$u'_{i}(x, t) = \frac{1}{\sqrt{N}} \sum_{k=1}^{N} X^{(k)}_{i}(x - x^k)$$

with,

$$X^{(k)}_{i} = a_{ij} \varepsilon_j f \sigma(x - x^k)$$

$X^k_i$ being independent random variables following the same distribution, the central limit theorem can be applied. This states that when $N$ tends towards infinity, the probability density function of $u'_{i}(x, t)$ tends towards a Normal distribution $N(\mu_i, \sigma^2_i)$ of mean $\mu_i = \langle X^k_i \rangle$ and of variance $\sigma^2_i = \langle (X^k_i)^2 \rangle$. In this method it would mean, when the number of eddies $N$ tends towards infinity, the signature of each eddy in the final synthetic signal becomes more faint and the final signal tends toward a universal Gaussian state.

From there we can thus express the skewness of the velocity signal using Eq.(3.20).

$$S_{u_i} = \frac{\langle u'^3_i \rangle}{\langle u'^2_i \rangle^{3/2}} = \frac{1}{(NR_{ii})^{3/2}} \left\langle \left( \sum_{k=1}^{N} X^{(k)}_{i} \right)^3 \right\rangle$$

which is shown to be zero

$$S_{u_i} = 0$$

Actually using the multinomial theorem,

$$(x_1 + x_2 + \ldots + x_m)^n = \sum_{k_1,k_2,\ldots,k_m} \frac{n!}{k_1!k_2!\ldots k_m!} x_1^{k_1} x_2^{k_2} \ldots x_m^{k_m}$$

with $\sum k_i = n$ for any positive integer $m$ and any non-negative integer $n$, we obtain,
3.2. SIGNAL CHARACTERISTICS

\[ S_{ui} = \frac{1}{(NR_{ii})^{3/2}} \sum_{k_1,k_2,\ldots,k_N} \frac{n!}{k_1!k_2!\cdots k_N!} \langle (X_i^{(1)})^{k_1} \rangle \langle (X_i^{(2)})^{k_2} \rangle \cdots \langle (X_i^{(N)})^{k_N} \rangle \]

(3.24)

which implies a nondegenerate sum only if the contributions to \( k_p \) of the form \( 3+0+0 \) are employed. (In fact \( \langle X_p^k \rangle = 0 \) if \( k_p = 1 \) since the average fluctuations generated by each eddy is zero).

Then using the independence of position and intensities, the third order moment of \( X_i^p \) is written as

\[ \langle (X_i^p)^3 \rangle = \langle (a_{ij}\varepsilon_j^k)^3 \rangle \langle (f_\sigma(x - x_k))^3 \rangle \]

(3.25)

whose factor \( \langle (a_{ij}\varepsilon_j^k)^3 \rangle \) can be expanded as following using again the multinomial theorem,

\[ \langle (a_{ij}\varepsilon_j^k)^3 \rangle = \sum_{k_1,k_2,k_3} \frac{3!}{k_1!k_2!k_3!} (a_{i1}\varepsilon_1^k)^{k_1}(a_{i2}\varepsilon_2^k)^{k_2}(a_{i3}\varepsilon_3^k)^{k_3} \]

(3.26)

Since by definition \( \langle \varepsilon^k \rangle = 0 \) and \( \langle (\varepsilon^k)^3 \rangle = 0 \) as the intensities are taken from a nonskewed distribution, the third order moment of \( X_i^p \) is zero and all the terms in the sum on Eq. (3.24) are zero.

3.2.3.2 Flatness

Finally we can express the flatness of the signal using Eq.(3.20)

\[ F_{ui} = \frac{\langle u_i^4 \rangle}{\langle u_i^2 \rangle^2} = \frac{1}{N^2 R_{ii}^2} \left\langle \left( \sum_{k=1}^{N} X_i^{(k)} \right)^4 \right\rangle \]

(3.27)

which is equal after simplification to, see Ref. [18],

\[ F_{ui} = 3 + \frac{1}{N} \left( 4F_3^2 F_e \frac{V_B}{\sigma^2} - 3 \right) \]

(3.28)
where $F_\varepsilon$ is the flatness of the PDF of $\varepsilon_k^j$. This is only valid when the Reynolds stress tensor is diagonal which interests us to validate the first case implementation in a HIT.

This expression shows the influence of the different parameters on the flatness of the velocity fluctuations. However this won’t be studied in detail but just used as a reference for the validation of the method which should give a flatness different from zero contrary to the skewness.

Since the velocity is based on intensities $\varepsilon$ taken from a non-skewed distribution, the odd moments of all linear combination will also vanish. This means that all the odd moment will be equal to zero whereas the even moment will be nonzero.

Note: The derivation of this last expression, based on the same principles as for the skewness, won’t be detailed in this report. (see Ref. [18] for more details).
Chapter 4

Results

Once the theory has been analyzed, the SEM code has been implemented. Firstly a simple case HIT defined as following, has been treated in order to validate only the algorithm. Time averaged statistics have been extracted from the simulations and have been compared with exact results derived in the previous section. Moreover parameters variations have shown the response of the method and its sensitivity allowing a better comprehension. After this first validation the code has been finally integrated in the elsA package respecting its environment.

4.1 Homogeneous Isotropic Turbulence (HIT)

4.1.1 Definition

A field of fluctuations as:

- homogeneous, if all its statistic properties are invariant for every translation.
- isotropic, if all its statistic properties are invariant to rotation and plane reflections. Therefore an isotropic field is necessarily homogeneous since any translation is the composition of two rotations.

The Reynolds stress tensor of HIT has all the diagonal elements are equal and the non-diagonal terms are zero as following:

\[ u'u' = v'v' = w'w' = \frac{2}{3}k \text{ and } u'v' = v'w' = u'w' = 0 \]

obtaining the Reynolds stresses tensor:

\[ R_{ij} = \begin{pmatrix} \frac{2}{3}k & 0 & 0 \\ 0 & \frac{2}{3}k & 0 \\ 0 & 0 & \frac{2}{3}k \end{pmatrix} \tag{4.1} \]

with k, the turbulent kinetic energy equal to
Thus this simplifies the implementation and validation exposed in the next section.

4.1.2 Simulation setup and parameters

In this previous step, the goal is not to check the method by observing the developing turbulence but only by analyzing the eddies generated.

The code has been firstly implemented in Python for simplicity reasons. However this algorithm has shown to be very costly in computational time for a language which is executed by an interpreter instead of a previous compilation before execution. Actually the algorithm is based on a double loop depending on the number of eddies on the box and on the mesh points, over matrix-matrix and matrix-vector multiplications. Despite the optimizations, the computing time has still been important which has lead to a translation into Fortran in order to obtain better performances. This last code has been also required for the final step, namely, the integration in the whole CFD package elsA.

The simulation has been done in a two-dimensional plane ($Oyz$) of size $2\pi \times 2\pi$. The eddies are generated every $dt=0.01\text{s}$ in the center of every $128 \times 128$ grid cells with a spatial step of $\Delta y = \Delta z \approx 0.1\sigma$. In the Fig. 4.1 we can see the plane surrounded by the eddies generated in the box. In fact this can be seen as passing a uniform stream through a grid in wind-tunnel experiments.

After defining the discretization, the simulation data has been chosen as following:

- The mean flow: $U_0 = 10\text{m/s}$ in the x-direction normal to the plane
- The rms velocity: $u'_0 = 1\text{m/s}$

Using the Reynolds stresses tensor previously defined (Eq. (4.1)) and the turbulent kinetic energy (Eq. (4.2)), it follows:

$$
\begin{pmatrix}
1 & 0 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1 \\
\end{pmatrix}
$$

(4.3)

which will simplify our problem as expected.

Moreover the parameters to change correspond to:

32
4.1. HOMOGENEOUS ISOTROPIC TURBULENCE (HIT)

Figure 4.1. Inlet plane of eddy generation, surrounded by a box of eddies. 128x128 two-dimensional grid. Simulation of isotropic turbulence with the SEM. Taken from Ref. [18]

- $N \in \{10, 100, 1000, 10000\}$ the number of eddies in the box
- $\sigma \in \{0.25, 0.5, 1\}$: the length scale
- $f$ which will be a tent or a step function

After defining the setup and the variables to modify we will firstly validate the method by statistics computations in order to continue with a study of the parameters influence in space and time simulations.

Remarks concerning the simulations run: The values by default when no specified will be:
- $N = 1000$
- $\sigma = 0.5$
- $f(x) = \begin{cases} \sqrt{\frac{3}{2}}(1 - |x|), & \text{if } x < 1 \\ 0, & \text{otherwise} \end{cases}$

Moreover the temporal analysis will be realized on the point $(x, y) = (\pi, \pi)$ and the spatial representation will correspond to the last iteration: 1000.

4.1.3 Statistics

As already mentioned we need firstly to check the validity of the method before any further study regarding neither the algorithm nor the application in a channel flow. Thus statistics output of simulations, see Fig. 4.2, will be compared to the theory
previously presented. As we can observe on the Fig. 4.2a, the time averaged $u_i$ converges almost instantaneously to the correct values, it means: $< u > = 10$ and $< v > = < w > = 0$, as well as the Reynolds stress tensor on the Fig. 4.2b converging to the initial one (cf. Eq. (4.3)) after 20s. Finally concerning the skewness (Fig. 4.2c) and the flatness (Fig. 4.2d), the computations show the expected value. The odd moment (here the skewness) tends to zero as in the Eq. (3.22) and the even moment, the flatness, is different from zero tending to 3.5 as in the Eq. (3.28). This last equation reminded as following reveals the strong dependence of this last moment with respect to the parameters selected, contrary to the skewness.

$$F_{u_i} = 3 + \frac{1}{N} \left( 4F^3 \frac{V_B}{\sigma^3} - 3 \right)$$

As we can see, the number of eddies N has a major influence on the signal flatness. As N goes to infinity the flatness tends towards its Gaussian value of 3. Therefore it will additionally avoid any influence coming from the eddy size. This is also true for N near zero, the flatness will grow to infinity independently of the length scale. Moreover, for a fix N, an increase of the ratio $\frac{V_B}{\sigma^3}$ between the volume of the box of eddies and the cube of the length scale, can also cause an important increment of the flatness with respect to the value 3. This would mean that the box of eddies has been insufficiently filled because of a too large box or a too small eddies size.

Then an optimized combination of both variables should be studied regarding the flatness required. However for computational reasons the value of N, which plays a major role, should be kept as small as possible.

We have to notice that the large fluctuations appearing at the beginning means that the averaging time is too small. We can compare this fact to the time averaged sinusoid as in Eq. (4.4)

$$\frac{1}{T} \int_0^T \sin(wt) dt \rightarrow O\left( \frac{1}{T} \right)$$

Once the implementation has been validated we check the influence of the parameters allowing us to understand better the functioning of the SEM algorithm.

4.1.4 Parameters influence: Spatial and temporal analysis

As expected, the Fig. 4.3 shows how the signal is stationary in time and homogeneous and isotropic in space. This computation corresponds to a balanced combination of the parameters with a number of eddies $N=1000$ and a medium size $\sigma = 0.5$. Then in the next simulations just one of the velocity components will be observed.

4.1.4.1 Eddies number(N)

Firstly we will study the effect of the number of eddies on the synthetic velocities generation. Here three cases will be presented on the Fig. 4.4 for $N=10$, $N=100$
4.1. HOMOGENEOUS ISOTROPIC TURBULENCE (HIT)

and N=10000 using a length scale $\sigma = 1$. As expected the less eddies are present, the more are the empty regions without any fluctuation as well in time than in space domain, whereas higher are the amplitude of the eddies. This is explained by the concentration of energy on the few eddies. In fact Jarrin [18] proves this idea through the probability density functions of the velocities which show that the energy of the signal does not depend on the number of eddies.

Moreover we can clearly observe in the case $N = 10$, the travel time of the eddy through the plane which corresponds to the period of time of the peak, known as the time scale $\tau = 2\sigma U$. However more the number of eddies is increased and more difficult is to discern these moments because of a higher frequency.

4.1.4.2 Length scale ($\sigma$)

Then we investigate the influence of the size of the eddies $\sigma$ by running some simulations for values of $\sigma = 0.25$, 0.5 and 1 shown on the Fig. 4.5. As expected the bigger $\sigma$ is, the larger the length and time scales of the signal are. However if one reduces the size of the eddies without adding more eddies inside of the box then the region of influence will be finer causing a higher irregularity and intermittence of the turbulent signal. Finally as for the previous case, we can observe the time scale $\tau = \frac{2\sigma}{U}$ which actually increases for bigger eddy sizes. However when $\sigma$ is small the frequency is much higher.
Figure 4.3. Spatial representation (left) and time-history of the velocity $u$, $v$, $w$ at point $(\pi, \pi)$ (right), for $N=1000$ and $\sigma = 0.5$.
4.1. HOMOGENEOUS ISOTROPIC TURBULENCE (HIT)

Figure 4.4. Spacial representation (left) and time-history of the velocity $u$ at point $(\pi, \pi)$ (right), for $N=10, N=100, N=10000$ from top to bottom and $\sigma = 1$.
Figure 4.5. Spacial representation of the velocity $w$ (left) and time-history of $v$ at point $(\pi, \pi)$ (right), for $\sigma = 0.25m, \sigma = 0.5m, \sigma = 1m$ from top to bottom and $N=1000$
4.1. HOMOGENEOUS ISOTROPIC TURBULENCE (HIT)

4.1.4.3 Distribution function

Finally in the Fig. 4.6 we can have an overview of the influence on the SEM due to the choice of the distribution function. For that we compare the two following shape functions:

- A tent function (left)
  \[ f(x) = \begin{cases} \sqrt{2}(1-|x|), & \text{if } x < 1 \\ 0, & \text{otherwise} \end{cases} \]

- A step one (right)
  \[ f(x) = \begin{cases} \frac{1}{\sqrt{2}}, & \text{if } x < 1 \\ 0, & \text{otherwise} \end{cases} \]

The results observed on the Fig. 4.6 are the consequence of the following fact: the signal using a step function contains more energy at high frequencies than the other signal. In fact the energy spectrums shown by Jarrin [18] confirm this point.

Since the shape function defines the eddy, we understand the change of shape, for the first case spherics and for the second cubic.

![Figure 4.6](image)

**Figure 4.6.** Influence of the shape functions in the SEM eddies generation. Spacial representation of the velocity v using a tent function (left) and a step function(right), for \( \sigma = 0.5 \) and N=1000

After having validated and understood the method for a HIT case it will be extended to a channel flow computation. Before inserting it into elsA, the previous code will be extended to a general case namely, a non-HIT, in order to check the new SEM behaviour and insert it into elsA.
4.1.5 Transition to a Non-HIT

In this section the code has been then applied to a general case as a first extension to a channel case. Now the code takes into account general input data namely the variation of the mean velocities and a Reynolds stress tensor corresponding to an inlet plane closed by two parallel walls as shown in Fig. 4.8a and 4.7 respectively. As we can see in the Fig. 4.7, the autocorrelation terms of the Reynolds stress tensor reveals the presence of walls by colors (the same phenomenon for the crosscorrelation terms), showing velocity layers, homogeneous along the walls and with zero value at the walls (cf. the velocity profiles in Fig. 4.7). (Remark: the orientation of the walls has been inverted to a vertical position in Fig. 4.7) In Fig. 4.8a we can observe the same behavior for the mean velocity U and after the SEM computation we obtain the synthetic velocities with the corresponding profiles of u,v,w as seen in Fig 4.8b. The homogeneous layers have been then disturbed by the eddies fluctuations and the new velocity profiles show as well fluctuations along the mean values. Then we notice clearly the presence of the walls compared to the HIT case with a generated turbulence along the walls in the streamwise direction x.

The SEM behaving as expected with walls, this code will be applied to a channel case in the elsA package.

![Figure 4.7. Autocorrelation terms of the Reynolds stress tensor](image-url)
4.2. CHANNEL FLOW COMPUTATIONS

4.2.1 Spatially Developing Channel Flow

4.2.1.1 A Modified SEM

Real reproduction of the non-isotropic nature of the turbulence in the near-wall region, requires a slight modification of the SEM compared to the previous chapter. Up to now it was assumed that the length scale $\sigma$ is the same for all velocity components and for all directions. However it has been shown that this is not the case and that streamwise elongated coherent structures dominate the near-wall region instead of isotropic structures [18]. Thus a different length scale $\sigma_{ij}$ is used for every eddy i of the plane but also for each direction j. Thus we can express:

- The box of eddies $B$

$$x_{j,\text{min}} = \min_{x \in S, j \in 1,2,3} (x_j - \sigma_{ij}(x)) \quad \text{and} \quad x_{j,\text{max}} = \max_{x \in S, j \in 1,2,3} (x_j + \sigma_{ij}(x))$$  \hspace{1cm} (4.5)

- Velocity signal

$$u_i = U_i + \frac{1}{\sqrt{N}} \sum_{k=1}^{N} c_i^k f_\sigma(x - x^k)$$ \hspace{1cm} (4.6)

with the shape function $f_\sigma(x - x^k)$ of the eddies on position i given by

$$f_{\sigma_{ij}}(x - x_k) = \sqrt{V_B} \frac{1}{\sigma_{i1}} f(x_1 - x^k_1) \frac{1}{\sigma_{i2}} f(x_2 - x^k_2) \frac{1}{\sigma_{i3}} f(x_3 - x^k_3)$$ \hspace{1cm} (4.7)
4.2.1.2 Implementation

In this section the modified SEM derived from the previous validated code will finally be integrated in elsA which is a CFD software package for numerical simulation of turbulent compressible flows.

The implementation has been then split into three parts:

- Python script: for the input parameters and channel configuration
- C++ methods: for the SEM initialization and call of the fortran subroutine
- Fortran subroutine: for the generation of the synthetic velocities per time step

As following the different sections will be discussed more in detail in order to understand the main roles and the overall functioning and interaction between these 3 sections.

Python elsA

elsA code uses a user interface to describe a computational case, to carry it out and extract the results from it, in particular for post-processing and analysis. This user interface uses an Object-Oriented (O-O) conception, like the elsA kernel, but is implemented in a different O-O language (Python) and uses a simpler set of concepts, limited to the description of CFD cases. The high level interface provides interface classes: these interface classes, whose instances are the description objects, completely define -along with a few functions- the possibilities of user communication with the kernel, that is the interaction model.

Then the script has been fragmented as following: The main file where the whole elsA code running is managed, defines firstly the CFD Problem, the model, the Numerics (in time and space), the viscous flux and imports the other files determining the boundary conditions, the topology, the parameters and the variables to extract. In the topology file the domain is divided into blocks linked to a processor. The type of the boundaries corresponding to the entry face and output face has then been modified as inlet and outlet respectively. In the parameters file the number of iteration and the time step can be managed. In the boundary conditions file the SEM input data $N$, $\sigma$ and the Reynolds stress tensor, given through a file name, has been added. In fact an additional file regarding the Reynolds stress terms has been generated from another DNS simulation. This has required firstly a symmetry computation of the values for the other half channel and also an interpolation to the finer domain of elsA as seen on the Fig. 4.9. Additionally a change of the data orientation respecting the new rotated axes has been necessary and finally a new dimensioning has been carried out applying a constant factor.

It is also required the integration of new KEY variables inside the elsA code in order to be interpreted by the Python scripts and used afterwards by the code in
4.2. CHANNEL FLOW COMPUTATIONS

Figure 4.9. Symmetry and interpolation of the input data namely the streamwise velocity $U$ and the autocorrelation terms of the Reynolds stress tensor

C++ and Fortran. Moreover the variables to extract after each computation have to be specified in the extract file. In our case the conservative variables and the rms (root mean square) values have been chosen.

Since the user interface is external to the elsA code, we understand the interest in specifying the SEM parameters in this part, avoiding a compilation for each test case and therefore saving time.

Once the input data variables have been integrated, the SEM algorithm has to be inserted into the whole computation.

**C++ file**

 elsA code has been constructed as a very deep and complex tree of folders of specific functionalities inside the CFD computation. Since the SEM algorithm works on the boundary conditions the modifications have mainly affected the files related to it. All the data initialization computations have been inserted and fragmented into methods which use the input SEM data sent to them as input parameters. In fact the interaction between the specified SEM data from the python and this SEM methods has been managed at another level in the folder tree where the SEM KEY variables values defined in the python file are extracted and sent to the previous
function as parameters. The reading of the SEM file related to the Reynolds stress tensor terms is also done at this level.

Then, once all the data has been initialized, the main SEM computation should be called. This part will also be exported at another level and in another language, Fortran, for time optimization reasons.

**Fortran file**

In this subroutine, the main Synthetic Eddy Method algorithm appears, namely the computation of the synthetic velocity which is returned as output value per time step. Actually this is another difference with respect to the first version since the loop over the time has been removed. The overall elsA code will manage it when computing the flow. For every time step it will also call the SEM methods and subroutines in order to use the new inflow boundary conditions.

An important point regarding the division of the code, concerns the implementation method of the shared variables. In fact for some cases their value don’t have to be lost by transmission and the addition of global variables in the .h files and a dynamic allocation memory have been required.

The following Fig. 4.10 summarizes the main idea, giving an overview of the general procedure of the whole elsA code and the interaction with the SEM.

### 4.2.1.3 Setup and Simulation

The computational domain dimensions is of $10\pi\delta \times 2\delta \times \pi\delta$ with a grid of $161 \times 101 \times 81$ cells in the streamwise, wall-normal and spanwise directions, respectively. Here the dimensionalization is done through $\delta$ which represents the half of the channel high. Periodic boundary conditions are applied in the spanwise directions, whereas no-slip boundary conditions are imposed at the walls. In streamwise direction inlet and outlet boundary conditions are set. Fig. 4.11 shows the whole setup of the simulation. SEM velocities are applied as inflow boundary conditions and at the outlet, the pressure is specified.

The following computations (Fig. 4.12 and 4.13) correspond to 65000 time steps thus to three full days of computations or one week of work since the simulations have to be run manually per 5000 iterations. We can then estimate the time per iteration equal to 4 seconds.

Concerning the configuration of the channel, the friction Reynolds number $Re_\tau = 395$ and the mean flow Reynolds number $Re = 6600$ with a WALE (Wall Adapting Local Eddy-Viscosity) model. Moreover the spatial discretization has been done with Finite Volume method and an implicit Backward Euler method
4.2. CHANNEL FLOW COMPUTATIONS

Figure 4.10. Integration of the Synthetic Eddy Method in elsA

has been used for the time discretization with a time step \( dt = 0.01s \). The SEM parameters chosen are a tent function for the shape function of the eddies and the number of eddies has been set to 10,000.

Then in the Fig. 4.12a we observe how the synthetic eddies injected on the inlet plane result on a physical turbulence along the channel and through the time steps. In fact this is clearly seen on the Fig. 4.12b where in the inlet plane eddy velocities are set along the wall as initial condition whereas in the outlet plane we observe the developed turbulent flow.

Fig. 4.13 shows another representation computing the iso-rotational of the velocity \( \mathbf{u} \).
Figure 4.11. Setup of the spatial channel flow using the SEM

Figure 4.12. Spatially Developing Channel Flow Computations. Velocity $u_z$
4.2. CHANNEL FLOW COMPUTATIONS

In order to validate the turbulence generated, an additional simulation has been run as following for a periodic channel case used as a reference.

4.2.2 Periodic Channel Flow

As shown in Fig. 4.14a, the channel flow simulations have been carried out in a computational domain of dimensions $2\pi\delta \times 2\delta \times \pi\delta$ with a grid of $81 \times 101 \times 81$ cells in the streamwise, wall-normal and spanwise directions, respectively. Periodic boundary conditions are applied in the streamwise and spanwise directions, whereas no-slip boundary conditions are imposed at the walls (cf. Fig. 4.14a) Which means that initially a turbulence will be applied in the channel and the flow will be re-injected iteratively in the streamwise and spanwise direction. A period of time or number of iterations will be necessary until reaching the convergence of statistics.

The simulations performed concern turbulent plane channel flow after 25000 time steps. At this time we can assume that the stationary step has been reached. Contrary to the first case, the spatially Developing Channel Flow, the channel can be reduced since the turbulence is developed in time and no in space anymore. Concerning the configuration of the channel, the same as for the Spatially Developing Channel Flow has been used and some simulations have been run.

In Fig. 4.14 we can actually observe the turbulent flow channel through different representations: the velocity $w$ in the streamwise direction (Fig. 4.14b), presented on the Fig. 4.14c and d by colour on the boundaries and as iso-surfaces, respectively. We consider that after this period of time, the process is stationary.
4.2.3 Statistics

In this section the first analysis of time averaged statistics extracted from the previous simulations are exposed. This first step of statistics study will determine the next steps to achieve as a future work.

Fig. 4.15 and 4.16 give a first overview of the rms (root mean square) of the streamwise velocity for the periodic and spatial case. These velocities profiles confirm the previous result in spatial domain where the periodic channel shows to have reached a stationary state contrary to the spatial one. In fact the periodic case could be seen as an equivalent infinite channel. However in the spatial channel an important increase of the fluctuations are shown after 5000 iterations. If we then compute the flow-through times as following:

\[ n = \frac{\tau NU}{L} \]

(4.8)

with the dimensionless values: \( \tau = 0.01, U = 1, L = 10\pi, N = 60000 \), we find a number of times around 20 which could explain a non reached stationary state.
4.2. CHANNEL FLOW COMPUTATIONS

Therefore we can also notice on the previous simulations how the turbulence in the middle of the channel seems to be more developed for the periodic LES computations than for the spatial channel.

These observations would require an increase of the iterations but also of the channel size. In order to check if the channel size is enough large, data extraction should be taken at two different points far in the downstream channel. Similar results extracted at these two points would validate the selected size.

However a loss of symmetry is observed contrary to the periodic case, which should be treated.

Finally, contrary to the HIT and as expected, we observe for both cases a zero velocity on the walls.

Thus these remarks show the need of a further study based on the channel size, an increase of the time steps number and on SEM parameter variations.

Figure 4.15. Periodic channel statistics for 25000 time steps
Figure 4.16. Spatial channel statistics. For 60000 time steps (a) and 65000 time steps (b)
Chapter 5

Discussion

In this project the Synthetic Eddy Method of generating inflow data for LES has been implemented in the elsA code. The method, based on the classical view of turbulence as a superposition of eddies has firstly been used in a HIT. The study of statistics has allowed to validate the method and the variation of the parameters has revealed the sensitivity of the procedure with respect to them, namely the number of eddies $N$, the length scale $\sigma$, the distribution function $f$. However the choice of these variables should not be taken independently from each other since the results obtained from every parameter value determined will depend on the configuration of the others. For example if we want to decrease the eddy size without loosing an homogeneous spatial distribution of the fluctuations, the number of eddies will have to be increase. However if we choose a big length scale $\sigma$ with respect to the dimension of the plane, the fluctuating signal will be smoother and this time the number of eddies won’t have any effect. This relation appears clearly through the influence on the flatness Eq. (3.28).

Then an extension has been possible to a more general and practical case, a channel flow computation. The simulation has also been run with a periodic LES as a reference of comparison for the SEM.

Even if physical turbulence have been observed, the periodic case shows higher turbulence which will require some improvements of the spatially developing channel flow case. Actually a further study can be done regarding the choice of the parameters, the size of the channel. Concerning the parameters, the eddy size should be limited by a fraction of the wall-normal dimension and the number of eddies $N$ as well as the channel dimensions by a computational cost. Statistics analysis already started will allow the validation with respect to the periodic LES reference.

Hence the Synthetic Eddy Method has shown to have several advantages as the elimination of a previous precursor simulation, its application to general flows and the control of the method by divers parameters defining the eddies. However this method is mainly based on random variables taking into account just the Reynolds stress tensor as physical contribution.

Therefore as a future work, even if the SEM has already been tested and vali-
dated in a wide variety of turbulent flows, further improvements of the algorithm could eventually been study as for instance the use of the shape function. Moreover comparisons with spectral methods could be carried out and a coupling with the RANS domain should be afterwards implemented.
Bibliography


