UNIVERSITÀ DI PISA

Università degli Studi di Pisa

FACOLTÀ DI INGEGNERIA
Corso di Laurea Magistrale in Ingegneria Aerospaziale

TESI DI LAUREA MAGISTRALE

Implementation and appraisal of a stochastic model for particle tracking in turbulent flow

Candidato:
Alessio Innocenti

Relatori:
Prof. Maria Vittoria Salvetti
Prof. Sergio Chibbaro
Prof. Cristian Marchioli

Anno Accademico 2014–2015

Abstract

The evaluation of particle concentration in wall-bounded turbulence is a problem for a variety of industrial and environmental applications. This problem cannot be tackled using a numerical approach based on Direct Numerical Simulation (DNS), which would require excessive computational costs to match flow Reynolds number and geometry configurations typical of practical flow instances. Currently, one of the most promising and viable approach is represented by Large-Eddy Simulation (LES), in which only the large flow scales are resolved directly taking into account the small flow scales (smaller than a given space filtering width) through a model. The missing scales in LES not only influence the resolved scales of the fluid flow, but also affect the motion of small particles present in the flow. The disregard of the smallest scales of fluid velocity in the particle motion leads to significant errors in the prediction of turbulent dispersion and of turbophoresis, the tendency of particles to accumulate at the walls in wall-bounded turbulent flows. The influence of the missing sub-grid scales can be modeled by a SGS model for particle equation.

In this work we adopted a stochastic model for the velocity seen by particles coupled with a LES solver for the fluid field. This model is an adaptation to the LES case of that proposed by Minier & Peirano [16] for RANS. We implemented and coupled it with a LES flow solver. Then, in the second part of the work, the mathematical validation carried out and the first results obtained with this model are shown. First of all, tests of consistency show that the model behaves well in the limit case of very small inertia particles, i.e. fluid particles. Duplicate fields obtained both from LES and from particles are compared. Finally tests with inertial particles show results in line with LES. This is a good result because it means that this class model, with some improvements, can really lead to a better description of the phenomenon addressed.

Contents

Abstract **i**

1 Introduction **1**

1.1 Two-phase flow regimes 2

1.2 Governing equations 3

1.3 Wall turbulence and coherent structures 4

2 Fluid models **8**

2.1 DNS approach 8

2.2 LES approach 9

2.3 PDF approach 11

2.3.1 Definitions and properties 11

2.3.2 The PDF transport equation 13

2.3.3 The Lagrangian approach 15

2.3.4 Stochastic differential equations 16

2.3.5 Shifting from an ordinary differential equation to a stochastic differential equation 16

2.3.6 Langevin equations 18

2.4 Velocity Filtered Density Function 23

3 Particle model **26**

3.1 Extension from the fluid model to the particle model 26

3.2 Limit cases 30

4 Numerical approach **33**

4.1 FLOWSB 35

4.1.1 Rephrased equations . . . 36
4.1.2 Channel flow equations on spectral space . . . 37
4.1.3 LES solver . . . 38
4.2 Weak numerical scheme for particles equation . . . 38
4.3 Limit systems of analytical solution . . . 43
4.4 Simulation of a Gaussian vector . . . 44
4.5 Weak first order scheme . . . 45
4.6 Weak second order scheme . . . 47

5 Implementation and validation **53**
5.1 Structure of FLOWSB . . . 53
5.2 Stochastic module . . . 54
5.3 Mathematical validation . . . 58
5.3.1 Preliminary study . . . 58
5.3.2 Limit cases . . . 59
5.4 Discretization error . . . 63
5.4.1 Analytical solutions . . . 66
5.4.2 Numerical study . . . 70

6 Results **73**
6.1 Test of consistency . . . 74
6.2 Test with inertial particles . . . 83
6.2.1 Simulation details . . . 83
6.2.2 Results . . . 84

7 Conclusions **91**

Appendices **93**

A Diffusion processes **94**

B Subroutines **99**

List of Figures

1.1 Two phase flow regimes on heat exchanger pipes. 3

1.2 Sweeps and ejections are some of the coherent structures forming near the wall and causing the particle accumulation. We represent particles moving away from wall in blue and particles moving toward the wall in red. 5

1.3 One low speed streak with counterrotating quasi-streamwise vortices. 6

1.4 Simultaneous presence of a 'mature' and a 'new' vortex that weaken the ejection like environment. 7

3.1 Crossing trajectory effect 27

4.1 Reference domain for LES channel flow. 35

5.1 Structure of the FLOWSB code 54

5.2 Structure of the subroutine parttrack 55

5.3 General case with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 2 \cdot 10^{-1}s$, $\tau_p = 10^{-1}s$, $\sigma = 10^{1}m/s^{3/2}$ 60

5.4 Limit case (i) with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 10^{-1}s$, $\tau_p = 10^{-5}s$, $\sigma = 10^{1}m/s^{3/2}$ 60

5.5 Limit case (ii) with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 10^{-5}s$, $\tau_p = 10^{-1}s$, $\sigma = 10^{3}m/s^{3/2}$ 61

5.6 Limit case (iii) with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 10^{-5}s$, $\tau_p = 2 \cdot 10^{-5}s$, $\sigma = 10^{3}m/s^{3/2}$ 61

5.7 Limit case (iv) with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 10^{-15}s$, $\tau_p = 10^{-1}s$, $\sigma = 10^{1}m/s^{3/2}$ 62

5.8 Behaviour of first order moments in different cases. 62

5.9 Study of the discretization error in the general case: $\alpha_0 = 0.5$, $a = 0.1$, $b = 0.25$, $\sigma_0 = 0.5$, $p = -1.2$. For sch1: $N = 3 \cdot 10^4$, $M = 100$, $T = 2.4s$. For sch2: $N = 9 \cdot 10^5$, $M = 200$, $T = 3.2s$. 71

5.10 Study of the discretization error in the limit case (i): $\alpha_0 = 0.5$, $a = 0.1$, $b = 250$, $\sigma_0 = 0.5$, $p = -1.2$. For sch1: $N = 3 \cdot 10^4$, $M = 100$, $T = 2.4s$. For sch2: $N = 9 \cdot 10^5$, $M = 200$, $T = 3.2s$. 71

5.11 Study of the discretization error in the limit case (ii): $\alpha_0 = 0.5$, $a = 200$, $b = 0.25$, $\sigma_0 = 50$, $p = -1.2$. For sch1: $N = 3 \cdot 10^4$, $M = 100$, $T = 2.4s$. For sch2: $N = 9 \cdot 10^5$, $M = 200$, $T = 3.2s$. 72

5.12 Study of the discretization error in the limit case (iii): $\alpha_0 = 0.5$, $a = 200$, $b = 250$, $\sigma_0 = 50$, $p = -1.2$. For sch1: $N = 3 \cdot 10^4$, $M = 100$, $T = 2.4s$. For sch2: $N = 9 \cdot 10^5$, $M = 200$, $T = 3.2s$. 72

6.1 Reynolds averaged quantities ($\Delta t^+ = 3000$) obtained with different particle model: no-interpolation of the LES and particles quantities, NGP and full interpolation of first order. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES. (a) particle density, (b) filtered streamwise velocity, (c) filtered streamwise velocity rms, (d) xx component of the subgrid stress tensor. 75

6.2 Reynolds averaged density ($\Delta t^+ = 3000$) normalized with particle position at the beginning of the time interval used to averaging. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES. 76

6.3 Reynolds averaged quantities ($\Delta t^+ = 3000$) plotted against LES and DNS results. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES. (a) filtered stream-wise velocity, (b) filtered stream-wise velocity rms. 77

6.4 Instantaneous velocity correlation. Scatter plot of velocity evaluated from LES and from VFDF at cell center: (a) stream-wise component, (b) wall normal component. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES. 77

6.5 Reynolds averaged density ($\Delta t^+ = 3000$) obtained in different test case: (a) test on $\langle U_f \rangle_L$, (b) test on pressure gradient. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES. . 78

6.6 Grid used for evaluating statistics with variable cell length: Δ_E is the length of the new grid centered on the nodes of the previous one, Δ is the length of the previous grid. 78

6.7 Reynolds averaged quantities ($\Delta t^+ = 3000$, $N_E = 40$) obtained for different cell length. (a) filtered streamwise velocity, (b) filtered streamwise velocity rms, (c) xx component of the subgrid stress tensor, (d) xz component of the subgrid stress tensor 79

6.8 Reynolds averaged quantities ($\Delta t^+ = 3000$, $\Delta_E = \Delta$) obtained for different number of particles per cell. (a) filtered streamwise velocity, (b) filtered streamwise velocity rms, (c) xx component of the subgrid stress tensor, (d) xz component of the subgrid stress tensor. 80

6.9 Reynolds averaged quantities ($\Delta t^+ = 3000$, $N = 40$) obtained with different fluid subgrid models for LES: no-model, smagorinsky and germano. (a) particle density, (b) filtered stream-wise velocity, (c) xx component of the subgrid stress tensor, (d) xz component of the subgrid stress tensor. 81

6.10 Fluid streaks in the near wall region. Low-speed (resp. high-speed) streaks are rendered using colored contours of negative (resp. positive) streamwise fluctuating velocity u'_x on a horizontal plane at $z^+ = 10$ from the wall. Panel (a) shows the streaky structures obtained from the LES; panel (b) shows the streaky structures from VFDF. LES is computed without subgrid model, and a number of particle per cell $N = 40$ is adopted. 82

6.11 Particle concentration obtained with DNS (DNS), LES without SGS modeling in the particle equation of motion (LES) and LES with stochastic model for particles (MODEL). Tests on a coarse computational grid of $32 \times 32 \times 33$ grid nodes ($CF8$). In panel (a) $St = 1$ particles, in panel (b) $St = 5$ particles and in panel (c) $St = 25$ particles. Concentrations are computed at $\Delta t_1^+ = t_1^+ - t_{START}^+ = 2130$ $[w.u.]$. . 85

6.12 Particle concentration obtained with DNS (DNS), LES without SGS modeling in the particle equation of motion (LES) and LES with stochastic model for particles (MODEL). Tests on a coarse computational grid of $32 \times 32 \times 33$ grid nodes ($CF8$). In panels (a)-(b) $St = 1$ particles, in panel (c)-(d) $St = 5$ particles and in panel (e)-(f) $St = 25$ particles. Concentrations are computed at $\Delta t_1^+ = t_1^+ - t_{START}^+ = 2130$ $[w.u.]$ on left panels and at $\Delta t_2^+ = t_2^+ - t_{START}^+ = 4320$ $[w.u.]$ on right panels. 87

6.13 Particle concentration obtained with DNS (DNS), LES without SGS modeling in the particle equation of motion (LES) and LES with stochastic model for particles (MODEL). Tests on a coarse computational grid of $32 \times 32 \times 65$ grid nodes ($CF8_4$). In panel (a) $St = 1$ particles, in panel (b) $St = 5$ particles and in panel (c) $St = 25$ particles. Concentrations are computed at $\Delta t_1^+ = t_1^+ - t_{START}^+ = 2130$ $[w.u.]$. . 88

6.14 Particle concentration obtained with DNS, LES with stochastic model for particles (MODEL), LES with stochastic model for particles and wall function for dissipation (MODEL-wall func.) and LES with stochastic model for particles and new computed dissipation (MODEL-dissip.). Tests on a coarse computational grid of $32 \times 32 \times 33$ grid nodes ($CF8$). In panel (a) $St = 1$ particles, in panel (b) $St = 5$ particles and in panel (c) $St = 25$ particles. Concentrations are computed at $\Delta t_1^+ = t_1^+ - t_{START}^+ = 2130$ $[w.u.]$. 90

List of Tables

4.1 Analytical solutions to system (3.2.1) for time-independent coefficients. 42
4.2 Derivation of the covariance matrix for constant coefficients. 46
4.3 Weak first-order scheme (Euler scheme) 48
4.4 Weak second-order scheme 52

5.1 Numerical simulation for system 5.3.1 in the general and limit cases. General case: $\Delta t \ll T$, τ_p. Limit case (i): $\tau_p \ll \Delta t \ll T$. Limit case (ii): $T \ll \Delta t \ll \tau_p$. Limit case (iii): T, $\tau_p \ll \Delta t$. Limit case (iv): $T \to 0$. 59
5.2 Analytical solutions to system (5.4.16) and analytical expressions for the second order moments of $U_s(t)$ and $U_p(t)$. 68
5.3 Analytical expressions for the second order moment of $x_p(t)$. 69
5.4 Numerical values of the parameters of system 5.4.16 in the general and limit cases. 70

6.1 Channel properties. 73
6.2 Fluid properties. 74
6.3 Particle properties. 83
6.4 Particle parameters. 83
6.5 Simulation parameters. 84

Chapter 1

Introduction

The physical problem addressed in this work is the dispersion of inertial particles in turbulent flows, which is characterized by macroscopic phenomena such as nonhomogeneous distribution, large-scale clustering, and preferential concentration due to the inertial bias between the denser particles and the lighter surrounding fluid. Both direct numerical simulation (DNS) and large-eddy simulation (LES) together with Lagrangian particle tracking (LPT), have been used to investigate and quantify the behavior of particles near the wall, for instance, in channel flow or in pipe flow. Among previous LES applications to gas-solid turbulent flows, the fluid SGS velocity fluctuations were neglected for particles tracking and previous works showed that this is not satisfactory for the prediction of particle accumulation at the wall. Vreman and Kuerten [2] showed that, due to both subgrid and modeling errors, LES underestimates the tendency of particles to move toward the wall by the effect of turbulence. To circumvent this problem Marchioli, Salvetti and Soldati [3], proposed a way to reconstruct fluid velocity fluctuations by fractal interpolation and approximate deconvolution techniques, but results showed that, even when closure models are able to recover the fraction of SGS turbulent kinetic energy for the fluid velocity field (not resolved in LES), prediction of local segregation and, in turn, of near-wall accumulation may still be inaccurate. Giving up with this kind of approach, further studies have been carried out by Bianco *et al.*[4] on the nature of the filtering error. Results showed that filtering error is stochastic and has a non-Gaussian distribution. In addition, the distribution of the filtering error depends strongly on the wall-normal coordinate being maximum in the buffer region. This findings established the requirements that any closure model aimed at recovering sub-grid scale effects on the dynamics of inertial particles must satisfy. These achievements led

us to think to a stochastic model for particle tracking. In particular we adopted a Langevin stochastic model to simulate the fluid velocity seen by particles inspired to the model proposed by Minier and Peirano [16] and adapted for the LES case, as it was originally written in a RANS context. Our purpose is thus to develop a stochastic model for the fluid velocity seen by particles and to couple it with a LES solver, trying in this way to restore the right amount of fluctuations and to obtain better results for how concern particles concentration at the wall.

1.1 Two-phase flow regimes

As it transpires from their name, two-phase flows are encountered when two non-miscible phases coexist. Depending on the form of the interface between the two media, different regimes can be found. This is illustrated in Figure 1.1 which shows a range of regimes for the case of a boiling liquid (for example water) in a classical heat exchanger. At the bottom of the tube, the liquid has not yet started to boil and we have a single-phase turbulent flow. When nucleation starts at the walls, bubbles can be found as separate inclusions within the liquid (bubbly flows). Then, as more vapour is created we go through the so-called slug and plug regimes where vapour occupies a more important volumetric fraction. Then, as the liquid continues to boil, we find the annular regime with a thin liquid layer at the walls and a central vapour flow with small droplets carried by the vapour. The wide variety of regimes, merely outlined above, is typical of immiscible liquid–gas or liquid–liquid flows since the interface can be deformed. Two of these regimes (the bubbly and annular regimes) are characterized by the presence of one phase, either liquid or vapour, as separate inclusions embedded in the other phase. These are two examples of what is defined as dispersed turbulent two-phase flows, where one phase (called the continuous phase) is a continuum and the other phase (called the dispersed phase) appears as separate inclusions dispersed within the continuous one, assumed here to be a turbulent fluid. When the dispersed phase is characterized by a distribution in size, one speaks of a *polydispersed turbulent two-phase flows*. The dispersed regime (either mono or polydispersed) is of first importance in most cases. It is always found when the dispersed phase is made up by solid particles (solid particles in a gas or a liquid turbulent flow). It is often found for a liquid dispersed as separate droplets in a gas flow (sprays for example) or for two immiscible liquids where one liquid is dispersed in the other liquid. In the present work we limit ourselves exclusively to the dispersed regime and we will talk of a fluid (the continuous phase) and of solid discrete particles. In particular, the second phase is characterized by small heavy spherical particles wich are carried by the continuous fluid

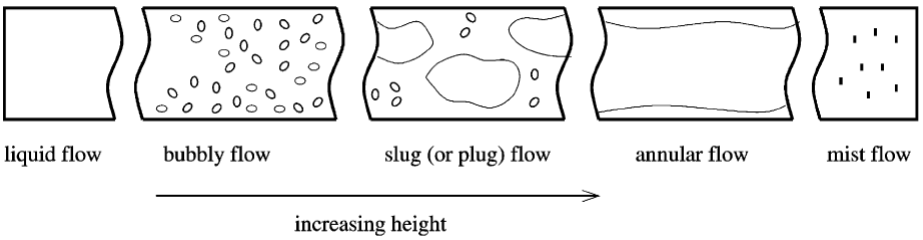


Figure 1.1: Two phase flow regimes on heat exchanger pipes.

phase in turbulent motion.

1.2 Governing equations

In this work, the dispersed phase will be assumed dilute; consequently, the one-way momentum coupling is adequate and particle collisions can safely be neglected. Yet, for a sufficiently high load of the dispersed phase, the two-way coupling needs to be accounted for in the momentum and energy equations; moreover, for high particle number densities, the interparticle collisions will affect their dynamics.

Additional complexity to the physical picture would be added through the interphase mass and energy transfer in the case of evaporating droplets or volatilizing solid particles. As told, no feedback from the dispersed particles to the carrying fluid is taken into account, thus the continuous phase will be resolved separately.

The continuous phase is supposed to be an incompressible turbulent fluid so that the governing motion equations are the continuity (which express mass conservation) and the Navier-Stokes equation (which express the linear momentum conservation) that, for an incompressible flow are:

$$\nabla \cdot \mathbf{U} = 0 \tag{1.2.1}$$

$$\rho \frac{D\mathbf{U}}{Dt} = -\nabla p + \mu \nabla^2 \mathbf{U} \tag{1.2.2}$$

where $\mathbf{U} = \mathbf{U}(\mathbf{x}, t)$ is the fluid velocity vector, $p = p(\mathbf{x}, t)$ is the equivalent pressure ($-\nabla p = -\nabla p + \rho \mathbf{g}$), ρ is the fluid density and μ the dynamic viscosity. These equations are numerically solved by using different methods that will be presented later.

The governing equation of the single particle of the second phase, see Maxey and Riley [5], will

be:

$$\frac{d\mathbf{X}}{dt} = \mathbf{U}_p \tag{1.2.3}$$

$$\begin{aligned}\frac{d\mathbf{U}_p}{dt} = \left(1 - \frac{\rho_f}{\rho_p}\right)\mathbf{g} + \frac{\mathbf{U}_s - \mathbf{U}_p}{\tau_p}(1 + 0.15Re_p^{0.687}) \\ + \frac{\rho_f}{\rho_p}\frac{D\mathbf{U}_p}{Dt} + C_L\frac{\rho_f}{\rho_p}[(\mathbf{U}_s - \mathbf{U}_p) \times \omega] \\ + \frac{9\mu}{d_p\rho_p\sqrt{\pi\nu}}\int_0^t \left(\frac{d\mathbf{U}_s}{dt} - \frac{d\mathbf{U}_p}{dt}\right)\frac{d\tau}{(t-\tau)^{0.5}} \\ + \frac{\rho_f}{2\rho_p}\left(\frac{D\mathbf{U}_s}{Dt} - \frac{d\mathbf{U}_p}{dt}\right)\end{aligned} \tag{1.2.4}$$

The first two terms on the right hand side are the buoyancy and the drag, while the other terms are in order the effect of the pressure gradient, the lift, the Basset term and the added mass term. Following the Lagrangian approach (or 'trajectory approach') the dispersed phase is directly treated as an ensemble of many individual inclusions; the exact instantaneous equations governing particle dynamics will be replaced by modelled ones for the reasons given above. As we will consider the case of heavy particles, so that the fluid density is order of magnitude lower than the particle density, the equation of the particle motion can be simplified further. It has been shown that with the assumption of $\rho_p/\rho_f \gg 1$ drag is the only significant force acting to the particle, so that the simplified equation that will be treated in this work is:

$$\frac{d\mathbf{U}_p}{dt} = \frac{\mathbf{U}_s - \mathbf{U}_p}{\tau_p}(1 + 0.15Re_p^{0.687}) \tag{1.2.5}$$

where $\mathbf{U}_s$ is the fluid velocity at the particle position and $\mathbf{U}_p$ is the particle velocity.

1.3 Wall turbulence and coherent structures

Particle transport, dispersion, and segregation in turbulent flows are highly non-uniform and intermittent phenomena which are recognized to depend on the local dynamics of turbulence structures. So a deep comprehension of the interactions between particle dynamics and turbulent transport and mixing is required. In literature, many papers (Marchioli *et al.*[6],[7]) have shown that there is a strong correlation among coherent wall structures, local particles segregation and resulting deposition phenomena. Modelling these physical mechanisms is not easy, especially with numerical methods coarser than DNS. The complication consists in the com-

plex interaction between particle inertia and the non-homogeneous structure of turbulence in the wall-normal direction and justifies the scarcity of physically-based accurate correlations for particle deposition flux. In the specific case of turbulent boundary layer, the local interaction between particles and turbulence structures leads to an unusually unique macroscopic behavior, i.e. particle accumulation in the viscous sublayer. This macroscopic behavior is caused from the combined action of the many microscopic transfer phenomena which drive particles toward the wall and away from the wall. In 1975 Cleaver and Yates [8] proposed a sub-layer model based on the Reynolds analogy for particle transport in turbulent boundary layer for the deposition of small solid particles from a gas stream. According to this mechanism, particles are driven toward the wall and away from the wall by *sweeps* (coherent downwash of outer fluid to the wall) and *ejections* (coherent upwash of wall fluid toward the outer flow) which are instantaneous realizations of the Reynolds stresses. In Figure 1.2 we represent schematically the situation.

Specifically, ejections bring the low-momentum fluid close to the wall into the outer region

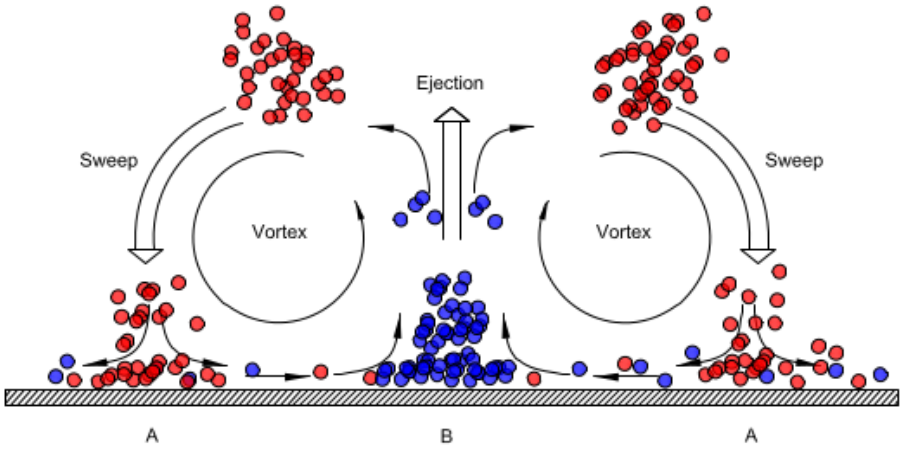


Figure 1.2: Sweeps and ejections are some of the coherent structures forming near the wall and causing the particle accumulation. We represent particles moving away from wall in blue and particles moving toward the wall in red.

whereas sweeps bring the high-momentum fluid from the outer flow into the wall region. From DNS studies, it results that particles segregating at the wall are many more than particles that ejections move away from the wall, because is more dificult to carry particles away from wall. To better understand the reasons of this apparently strange behaviour it's necessary to deeply investigate the mechanisms governing the dynamics of these coherent structures. Sweeps and ejections are in fact just the final outcome of the dynamics of turbulence structures in the wall

layer, and there is still some uncertainty about the mechanisms which generate and maintain the sweep/ejection events. They appear to be generated by the quasi-streamwise vortices which populate the near wall region. Quasi-streamwise vortices have been the object of a number of works. There is a general agreement about their characteristics: quasi-streamwise vortices appear to have a characteristic length of about 200 wall units and a spacing of about 400 wall units. These vortices are slightly tilted away from the wall and also with respect to flow direction. Owing to the continuous action of the quasi-streamwise vortices in generating sweeps and ejections, regions between two vortices are characterized by a streamwise velocity lower than the mean, whereas the regions outside the two vortices are characterized by a streamwise velocity higher than the mean. Specifically, the regions with velocity lower than the mean are called *low-speed streaks*, whereas the regions with velocity higher than the mean are called *high-speed regions*. Many quasi-streamwise vortices are usually associated with one single low-speed streak.

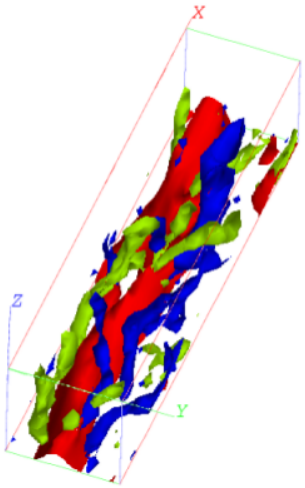


Figure 1.3: One low speed streak with counterrotating quasi-streamwise vortices.

In Figure 1.3 granted by Marchioli [9], a 450 wall units long piece of one low-speed streak is shown, flanked by two counter-rotating quasi-streamwise vortices. It has been shown that the generation of the quasi-streamwise vortices is associated with changes in the shape of the low-speed streak surface. In particular Schoppa & Hussain [10] showed from a stability analysis of an idealized low-speed streak that it is unstable to lateral perturbation. This lateral, sinuous instability is thus closely linked to the generation of new quasi-streamwise vortices which in turn generate sweeps and ejections that contribute to mantain the low-speed streak.

Results (Soldati & Marchioli [11]) show that clockwise and counterclockwise quasi-streamwise vortices appear flanking the low-speed streak as a staggered array in most of the cases. Only rarely do a clockwise and a counterclockwise quasi-streamwise vortices appear together. This peculiarity can be used to explain the reasons of particle segregation near the wall. We can envision the following cycle for particles initially in the outer flow: if a particle is captured by a sweep, it moves along a curved trajectory around the quasi-streamwise vortex generating the sweep, approaches the wall and moves between the vortex and the wall. During this phase, the particle may touch the wall or not. Then, the particle is on the upwash side of the vortex and is subject to the influence of the ejection. The next step involves trespassing the lifted low-speed streak and exiting from the wall layer.

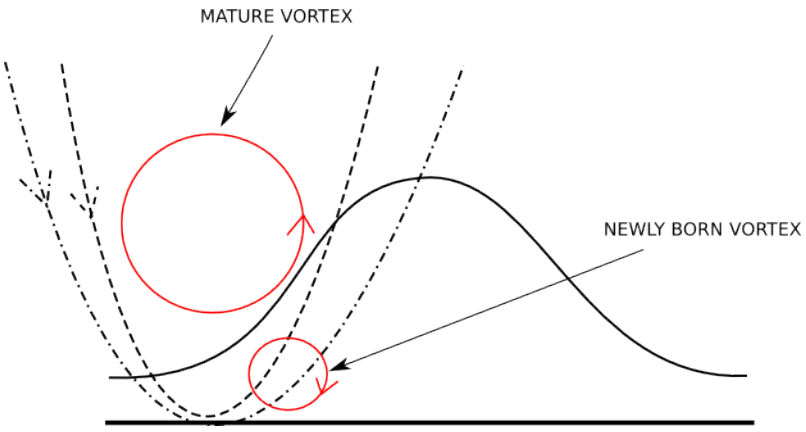


Figure 1.4: Simultaneous presence of a 'mature' and a 'new' vortex that weaken the ejection like environment.

Particles should migrate toward the surface of the lifted low-speed streak, which is an ejection-like environment, and find an ejection strong enough to drive them into the outer flow. Yet most of the particles remain trapped under the lifted low-speed streak. This may be explained with the fact that the two counterrotating quasi-streamwise vortex are staggered and do not appear at the same time. When a 'mature' one is trying to carry particles away from the wall, the co-presence of a newly born vortex of the opposite sign below the low speed streak, as illustrated in Figure 1.4, weaken the ejection so that not all particles carried to the wall by the sweep are able to go back to center of the channel.

This is just a possible explaination of particle transfer and segregation at the wall since this mechanism is not enterily clear, while the only certain thing is the segregation itself.

Chapter 2

Fluid models

2.1 DNS approach

With the simplifying hypothesis for which the two phases can be simulated separately, we start by introducing some of the most important techniques used to solve the equations of the continuous phase. The most precise and reliable methodology for the solution of Navier-Stokes equation is the direct numerical simulation. This means that the computational domain is divided in many parts by a three dimensional grid. The size of the mesh must be small enough to describe all the significative turbulent scales. Following the Kolmogrov cascade theory the mechanical energy is provided at the integral scale L (dimension comparable with the dimension of the domain) to the fluid by external processes and it is transfered scale by scale up to the dissipation ones (or Kolmogrov scale η) . Here, energy is dissipated by the viscous friction into heat.

This is an enormous problem when we consider turbulent flows at high Reynolds number because, the number of grid points must be sufficient to capture the wide range of turbulent structures, moreover the equation must be integrated on time considering discrete time-steps small enough to describe the faster velocity fluctuation related with the Kolmogrov scales. The dimension of the dissipative scales (η) and its characteristic frequency (f_η) are related with Reynolds number and for the turbulent channel flow the relations can be estimated as follow, see Pope [12]:

$$\begin{aligned} \Delta_x, \Delta_y, \Delta_z \leq \eta \approx l_0 Re_l^{-\frac{3}{4}} \\ \frac{1}{\Delta_t} \geq f_k \approx \nu l_0^{-2} Re_l^{\frac{3}{2}} \end{aligned} \tag{2.1.1}$$

where l_0 is the reference large scale length which can be approximately be considered as h which is the distance between the channel walls. Re_l is the Reynolds number and ν the fluid viscosity. As equation (1.2.1) and equation (1.2.2) fully describe the turbulent motion of incompressible Newtonian fluid, they are rewritten on discrete form and than computationally integrated on time considering enough point grids (L_x/Δ_x, L_y/Δ_y, $2h/\Delta_z$) and a sufficient computational time-step (Δ_t). Thus, the number of degree of freedom is very high and increases much more when we also have to simulate together the dispersed phase.
Direct numerical simulations are then possible in theory but very often impossible in practice. This method is limited to low Reynolds numbers and very simple geometries thus, flows of industrial interest cannot be studied. In any case the statistics and the physical insights that can be obtained by DNS are used to estimate the performances of other models. For this reason, this methodology is gaining progressively an attention equal to experimental results.
Most turbulent flows involve far too many degrees of freedom to be directly simulated. The issue is therefore to reduce the number of degrees of freedom to a tractable number and to come up with a contracted description.

2.2 LES approach

A less accurate approach, for the solution of Navier-Stokes equations is the so called Large Eddy Simulation or LES. In large-eddy simulation, the larger three-dimensional unsteady turbulent motions are directly represented, whereas th effects of the smaller scale motions are modelled. Nearly all of the computational effort in DNS is expended on the smallest, dissipative motions, whereas the energy and anisotropy are contained predominantly in the larger scales of motion.
In LES, the dynamics of the larger scale motions (which are affected by the flow geometry and are not universal) are computed explicitly, the influence of the smaller scales (which have, to some extent, a universal character) being represented by simple models. Thus, compared with DNS, the vast computational cost of explicitly representing the small-scale motions is avoided.
If we introduce a filtering operation, see Pope [12]:

$$\langle U(\boldsymbol{x},t)\rangle_L = \int_\Omega G(\boldsymbol{r},\boldsymbol{x})U(\boldsymbol{x}-\boldsymbol{r},t)d\boldsymbol{r} \tag{2.2.1}$$

where Ω is the flow domain and $G(\boldsymbol{r}, \boldsymbol{x})$ is the filter function and $U(\boldsymbol{x}, t)$ the quantity that has to be filtered we can decompose the velocity $\mathbf{U}(\mathbf{x}, t)$ into the sum of a filtered (or resolved) component $\langle \mathbf{U}(\mathbf{x}, t) \rangle_L$ and a residual (or subgrid-scale) component $\mathbf{u}'(\mathbf{x}, t)$. In this case $\langle \mathbf{U}(\mathbf{x}, t) \rangle_L$ (three dimensional and time dependent) represent the motion of the large eddies.
If we consider a generic filter operator $\langle\ \rangle_L$ that decompose a generic quantity in filtered and residual, the filtered continuity and Navier-Stokes equations can be rewritten as follow:

$$\frac{\partial \langle U_j \rangle_L}{\partial x_j} = 0 \tag{2.2.2}$$

$$\frac{\partial \langle U_i \rangle_L}{\partial t} + \langle U_j \rangle_L \frac{\partial \langle U_i \rangle_L}{\partial x_j} = -\frac{1}{\rho} \frac{\partial \langle p \rangle_L}{\partial x_i} + \nu \frac{\partial^2 \langle U_i \rangle_L}{\partial x_j \partial x_j} - \frac{\partial \tau_{ij}}{\partial x_j}. \tag{2.2.3}$$

where $\tau_{ij} = \langle U_i U_j \rangle_L - \langle U_i \rangle_L \langle U_j \rangle_L$ is the the SGS stress term that must be modelled to close the equation. The closure problem is one of the central problems in LES. The most commonly used SGS models are the Smagorinsky model and its variants. They model the unresolved turbulent scales, in analogy with the Bousinnesq hypothesis, through the addition of an eddy viscosity into the governing equations. The basic formulation of the Smagorinsky model is:

$$\tau_{ij} - \frac{\delta_{ij}}{3} \tau_{kk} = -2\nu_T \langle S_{ij} \rangle_L \tag{2.2.4}$$

where:

$$\langle S_{ij} \rangle_L = \frac{1}{2} \left(\frac{\partial \langle U_i \rangle_L}{\partial x_j} + \frac{\partial \langle U_j \rangle_L}{\partial x_i} \right) \tag{2.2.5}$$

is an entry of the strain rate tensor and the eddy viscosity ν_t is calculated as:

$$\nu_T = (C_s \Delta_g)^2 \sqrt{2 \langle S_{ij} \rangle_L \langle S_{ij} \rangle_L} \tag{2.2.6}$$

where Δ_g is the grid size and C_s is a constant. Many techniques have been developed to calculate C_s. Some models use a static value for C_s, often calculated from experiments of similar flows to those being modeled. Other models dynamically calculate C_s (dynamic Germano model) as a function of space and time. In some case, as usually happen in two-phase flow LES, the SGS term is completely neglected with the assumption that the long term behavior of the heavy particles is merely affected by the large scale structure.
Most commonly the filter used is defined in the spectral space as in our case, so that it operate

a cut-off of the highest wave numbers as follows:

$$\hat{G_\alpha}(k) = \begin{cases} 1 \; if \; -\pi/\Delta_\alpha \leq k \leq \pi/\Delta_\alpha \\ 0 \; otherwise \end{cases} \tag{2.2.7}$$

There are two sources of error with this kind of filtering (and with LES in general): an error due to the action of filtering itself, that means that only the largest scale (lowest wave number) are resolved, and a secondary source of error due to the model used for the closure of the equation which leads an error also in the resolved scales.

2.3 PDF approach

In this chapter we introduce also the PDF methods even if they are not used in our work for the fluid phase, but for the particles tracking. Anyway this approach was born for single phase fluid, so it's useful to understand the basis of this method for the description of the fluid only and to extend it to two phase flows in the next chapter. For a complete picture on this class of methods, see Pope [12]
In PDF methods, a model transport equation is solved for a PDF (probability density function) such as $f(\mathbf{V}; \boldsymbol{x}, t)$. In fact in a turbulent flow the velocity field is a random field so that it can be treated with a statistical approach. That $\mathbf{U}$ is a random variable means only that it does not have a unique value if one repeats an experiment under the same set of condition N times. This is because the extreme sensibility to initial and boundary condition of turbulent flows.

2.3.1 Definitions and properties

$f(\mathbf{V}; \boldsymbol{x}, t)$ is the one-point, one-time, Eulerian PDF of the velocity $\mathbf{U}(\boldsymbol{x}, t)$, where $\mathbf{V} = \{V_1, V_2, V_3\}$ is the indipendent variable in the sample space. Integration over the entire velocity space gives the normalization condition

$$\int f(\mathbf{V}; \boldsymbol{x}, t) d\mathbf{V} = 1 \tag{2.3.1}$$

Then for any function $Q(\mathbf{U}(\boldsymbol{x}, t))$, its mean is defined by

$$\langle Q(\mathbf{U}(\boldsymbol{x}, t)) \rangle = \int Q(\mathbf{V}) f(\mathbf{V}; \boldsymbol{x}, t) d\mathbf{V} \tag{2.3.2}$$

so that the mean velocity and the Reynolds stresses are given by

$$\langle \mathbf{U}(\boldsymbol{x},t)\rangle = \int \mathbf{V} f(\mathbf{V};\boldsymbol{x},t) d\mathbf{V} \tag{2.3.3}$$

and

$$\langle u_i u_j\rangle = \int (\mathbf{V}_i - \langle U_i\rangle)(\mathbf{V}_j - \langle U_j\rangle) f d\mathbf{V} \tag{2.3.4}$$

We define also the fine-grained PDF as follows

$$f'(\mathbf{V};\boldsymbol{x},t) = \delta(\mathbf{U}(\boldsymbol{x},t) - \mathbf{V}) = \prod_{i=1,3} \delta(U_i(\boldsymbol{x},t) - V_i) \tag{2.3.5}$$

At each point and time, f' is a three dimensional delta function in velocity space. The fine-grained PDF is very useful in obtaining and manipulating PDF equations because of the following two properties:

$$\langle f'(\mathbf{V};\boldsymbol{x},t)\rangle = f(\mathbf{V};\boldsymbol{x},t) \tag{2.3.6}$$

$$\langle \Phi(\boldsymbol{x},t) f'(\mathbf{V};\boldsymbol{x},t)\rangle = \langle \Phi(\boldsymbol{x},t)|\mathbf{U}(\boldsymbol{x},t) = \mathbf{V}\rangle f(\mathbf{V};\boldsymbol{x},t) \tag{2.3.7}$$

We can easily obtain these two properties simply using equation 2.3.2.
Finally we introduce the temporal and spatial derivatives of the fine-grained PDF. The derivative of the delta function $\delta(v-a)$ (with a being a constant) is denoted by $\delta^{(1)}(v-a)$ and it is an odd function

$$\frac{d}{dv}\delta(v-a) = \delta^{(1)}(v-a) = -\delta^{(1)}(a-v) \tag{2.3.8}$$

Thus, differentiating the fine grained PDF we obtain:

$$\frac{\partial}{\partial t} f'(\mathbf{V};\boldsymbol{x},t) = -\frac{\partial f'(\mathbf{V};\boldsymbol{x},t)}{\partial V_i}\frac{\partial U_i(\boldsymbol{x},t)}{\partial t} \tag{2.3.9}$$

$$\frac{\partial}{\partial x_i} f'(\mathbf{V};\boldsymbol{x},t) = -\frac{\partial f'(\mathbf{V};\boldsymbol{x},t)}{\partial V_j}\frac{\partial U_j(\boldsymbol{x},t)}{\partial x_i} \tag{2.3.10}$$

A final result required in the derivation of the PDF transport equation is

$$\begin{aligned} U_i(\boldsymbol{x},t)\frac{\partial}{\partial x_i} f'(\mathbf{V};\boldsymbol{x},t) &= \frac{\partial}{\partial x_i}[U_i(\boldsymbol{x},t) f'(\mathbf{V};\boldsymbol{x},t)] \\ &= \frac{\partial}{\partial x_i}[V_i f'(\mathbf{V};\boldsymbol{x},t)] = V_i \frac{\partial}{\partial x_i} f'(\mathbf{V};\boldsymbol{x},t) \end{aligned} \tag{2.3.11}$$

which can be obtained thanks to the incompressibility relation.

2.3.2 The PDF transport equation

From equations 2.3.9-2.3.11 we can obtain the substantial derivative of $f'(\mathbf{V};\boldsymbol{x},t)$

$$\frac{Df'}{Dt} = \frac{\partial f'}{\partial t} + V_i \frac{\partial f'}{\partial x_i} = -\frac{\partial}{\partial V_i}\Big(f' \frac{DU_i}{Dt}\Big) \tag{2.3.12}$$

and the mean of this equation yields

$$\frac{\partial f}{\partial t} + V_i \frac{\partial f}{\partial x_i} = -\frac{\partial}{\partial V_i}\Big(f \Big\langle \frac{DU_i}{Dt} \Big| \mathbf{V} \Big\rangle\Big) \tag{2.3.13}$$

This PDF equation is quite general and contains no specific physics (we have only used the incompressibility of the fluid). The physics enter when the Navier-Stokes equations are used to substitute for DU_i/Dt :

$$\frac{\partial f}{\partial t} + V_i \frac{\partial f}{\partial x_i} = -\frac{\partial}{\partial V_i}\Big(f \Big\langle \nu \nabla^2 U_i - \frac{1}{\rho}\frac{\partial p}{\partial x_i} \Big| \mathbf{V} \Big\rangle\Big) \tag{2.3.14}$$

The Reynolds decomposition of pressure ($p = \langle p \rangle + p'$) leads to

$$\Big\langle \frac{\partial p}{\partial x_i} \Big| \mathbf{V} \Big\rangle = \frac{\partial \langle p \rangle}{\partial x_i} + \Big\langle \frac{\partial p'}{\partial x_i} \Big| \mathbf{V} \Big\rangle \tag{2.3.15}$$

Being $\partial\langle p\rangle/\partial x_i$ non-random, it is unaffected by the mean and conditional mean operations, so that we finally obtain:

$$\frac{\partial f}{\partial t} + V_i \frac{\partial f}{\partial x_i} = \frac{1}{\rho}\frac{\partial \langle p \rangle}{\partial x_i}\frac{\partial f}{\partial V_i} - \frac{\partial}{\partial V_i}\Big(f \Big\langle \nu \nabla^2 U_i - \frac{1}{\rho}\frac{\partial p'}{\partial x_i} \Big| \mathbf{V} \Big\rangle\Big) \tag{2.3.16}$$

The terms on the left-hand side are in closed form as well as the mean pressure gradient. In fact, known $f(\mathbf{V};\boldsymbol{x},t)$ we can evaluate the mean velocity and the Reynolds stresses thanks to which we can evaluate the mean pressure from the poisson equation. Conversely the dissipative term and the gradient of the fluctuating pressure are not in closed form. The quantity $\nu\nabla^2 U_i$ is not known because we have the single-point PDF while to evaluate a derivative we would need the two-point one. For the same reason is not known the fluctuating pressure gradient, as the pressure is a global quantity which depends on the velocity in the whole field.
An equation like the 2.3.16 is called **Liouville** equation. There is always a perfect duality between

an equation which describes a dynamical stochastic system and a PDF transport equation:

$$\frac{D\mathbf{u}}{Dt} = \mathbf{A}(\boldsymbol{x}, \mathbf{u}) \tag{2.3.17}$$

$$\frac{\partial f}{\partial t} + V_i \frac{\partial f}{\partial x_i} = -\frac{\partial}{\partial V_i}\Big(f\Big\langle \mathbf{A}\Big|\mathbf{V}\Big\rangle\Big) \tag{2.3.18}$$

Once the PDF transport equation is resolved, we can evaluate for example the mean velocity $\langle \mathbf{U}(\boldsymbol{x}, t)\rangle$ which is the same quantity resolved in the RANS approach, even if the PDF approach is more accurate because the terms to be modelled for the closure of the equations are evidently less important than in the RANS case (convective terms).

Different types of models have been proposed for the closure of the PDF transport equation whereas the most commonly adopted is the generalized **Langevin** model which has the following form:

$$\frac{\partial f}{\partial t} + V_i \frac{\partial f}{\partial x_i} - \frac{1}{\rho}\frac{\partial \langle p\rangle}{\partial x_i}\frac{\partial f}{\partial V_i} = -\frac{\partial}{\partial V_i}[f G_{ij}(V_j - \langle U_j\rangle)] + \frac{1}{2}C_0\epsilon\frac{\partial^2 f}{\partial V_i \partial V_i} \tag{2.3.19}$$

where $C_0(\boldsymbol{x}, t)$ and $G_{ij}(\boldsymbol{x}, t)$ are coefficients that define the particular model: C_0 is non-dimensional, whereas G_{ij} has dimensions of inverse time. Here we limit ourselves to consider the simplest choice of these constants, which is the *simplified Langevin model*:

$$G_{ij} = -\Big(\frac{1}{2} + \frac{3}{4}C_0\Big)\frac{\epsilon}{k}\delta_{ij} \tag{2.3.20}$$

which correspond to the Rotta model in terms of Reynolds equation with

$$C_R = 1 + \frac{3}{2}C_0 \tag{2.3.21}$$

In fact from the PDF equation it's not hard to find the corresponding Reynolds equation by simply multiplying the equation by V_k and integrating. So each Langevin model has a corresponding Reynolds stress model. The constant G_{ij} cannot be assigned arbitrarly, but it must satisfy a condition dictated by the pressure-rate-of-strain tensor to be redistributive, which gives

$$(1 + \frac{3}{2}C_0)\epsilon + G_{ij}\langle u_i u_j\rangle = 0 \tag{2.3.22}$$

2.3.3 The Lagrangian approach

To better understand the meaning of the Langevin model and where does it come from, it's meaningful to show the Lagrangian approach, which can better give a physical meaning to the model. When Lagrangian approach is adopted, every fluid particle is followed so that the quantities attached to each particle can be denoted as $\mathbf{X}^+(t,\mathbf{Y})$ and $\mathbf{U}^+(t,\mathbf{Y})$ and they indicate the position and the velocity of the fluid particle originating from position $\mathbf{Y}$ at the reference time t_0. Their joint PDF is accordingly called the Lagrangian PDF and is denoted by $f_L(\mathbf{V},\boldsymbol{x};t|\mathbf{Y})$. As we can easily see, the difference between the Eulerian PDF is that the position in the sample space has become a dependent variable, and the PDF is conditioned to the initial position $\mathbf{Y}$. The Lagrangian PDF can be expressed in terms of the fine-grained Lagrangian PDF ${f'}_L$ as

$$f_L(\mathbf{V},\boldsymbol{x};t|\mathbf{Y}) = \langle {f'}_L(\mathbf{V},\boldsymbol{x};t|\mathbf{Y})\rangle \tag{2.3.23}$$

where

$${f'}_L(\mathbf{V},\boldsymbol{x};t|\mathbf{Y}) = \delta(\mathbf{U}^+(t,\mathbf{Y})-\mathbf{V})\delta(\mathbf{X}^+(t,\mathbf{Y})-\boldsymbol{x}) \tag{2.3.24}$$

Now, considering the integral of ${f'}_L$ over all initial points, since there is a one-to-one mapping between points $\mathbf{Y}$ and $\mathbf{X}^+(t,\mathbf{Y})$, we can express it as

$$\int {f'}_L d\mathbf{Y} = \int {f'}_L \mathbf{J}^{-1} d\mathbf{X}^+ \tag{2.3.25}$$

where $\mathbf{J}$ is the determinant of the Jacobian. For the incompressible flow being considered, a consequence of the continuity equation is that $\mathbf{J}$ is unity. Thus we have

$$\begin{aligned}\int {f'}_L d\mathbf{Y} &= \int \delta(\mathbf{U}^+(t,\mathbf{Y})-\mathbf{V})\delta(\mathbf{X}^+(t,\mathbf{Y})-\boldsymbol{x})d\mathbf{X}^+ \\ &= \delta(\mathbf{U}^+(t,\mathbf{Y})-\mathbf{V})|_{\mathbf{X}^+(t,\mathbf{Y})=\boldsymbol{x}} \\ &= \delta(\mathbf{U}(\boldsymbol{x},t)-\mathbf{V})\end{aligned} \tag{2.3.26}$$

The sifting property of the delta function singles out the fluid particle located at $\mathbf{X}^+(t,\mathbf{Y})=\boldsymbol{x}$, which has velocity $\mathbf{U}(\boldsymbol{x},t)$. It may be recognized that $\delta(\mathbf{U}(\boldsymbol{x},t)-\mathbf{V})$ is the fine-grained Eulerian PDF, so that the expectation of Eq. 2.3.26 leads to

$$\int f_L(\mathbf{V},\boldsymbol{x};t|\mathbf{Y})d\mathbf{Y} = f(\mathbf{V};\boldsymbol{x},t) \tag{2.3.27}$$

2.3.4 Stochastic differential equations

In the Lagrangian approach we will be concerned with the equations which describe particle dynamics, so Navier-Stokes equations written in the Lagrangian form

$$\frac{D\mathbf{U}^+}{Dt} = -\frac{1}{\rho}\nabla p^+ + \nu\nabla^2\mathbf{U}^+ \tag{2.3.28}$$

and

$$\frac{D\mathbf{X}^+}{Dt} = \mathbf{U}^+(t) \tag{2.3.29}$$

where the plus means that the quantities observed are Lagrangian, so attached to each particle. If these equations are directly used to obtain the PDF transport equation, Eq. 2.3.16, as seen in the section above, this equation is unclosed and a closure model is needed. However trying to model directly the PDF transport equation is difficult and far from the physical meaning. So the better way to proceed is to follow the Lagrangian approach and to model directly the Lagrangian Navier-Stokes equations. So the equation for the velocity becomes

$$\frac{D\mathbf{U}^*(t)}{Dt} = F[t, \mathbf{X}^*(t), \mathbf{U}^*(t)] \tag{2.3.30}$$

where with $\mathbf{U}^*(t)$ we indicate the modelled velocity. The principles of modelling is to replace *fast variables*, so those with very small characteristic times, with white noise processes (see Appendix A). This approach will lead to a stochastic differential equation for the velocity

$$d\mathbf{U}^*(t) = a[\mathbf{U}^*(t), t]dt + b[\mathbf{U}^*(t), t]d\mathbf{W}(t) \tag{2.3.31}$$

2.3.5 Shifting from an ordinary differential equation to a stochastic differential equation

Let us consider the case of a system $X(t)$ whose time rate of change is $Y(t)$

$$\frac{dX(t)}{dt} = Y(t) \tag{2.3.32}$$

We consider that we are dealing with stochastic processes which are differentiable and can those be handled with normal calculus rules. This gives

$$\frac{d\langle X^2(t)\rangle}{dt} = 2\int_0^t \langle Y(t)Y(t')\rangle dt' \tag{2.3.33}$$

Now for the sake of simplicity we consider $Y(t)$ as a stationary process and we introduce its autocorrelation $R_y(s)$ defined by $R_y(s) = \langle Y(t)Y(t+s)\rangle / \langle Y^2\rangle$, we can write

$$\frac{d\langle X^2(t)\rangle}{dt} = 2\langle Y^2\rangle \int_0^t R_y(s)ds \tag{2.3.34}$$

The important scale in that reasoning is the integral time scale of $Y(t)$, say T, which is defined by the integral of the autocorrelation

$$T = \int_0^\infty R_y(s)ds \tag{2.3.35}$$

This time scale is a measure of the 'memory' of the process. If we consider time intervals s small with respect to T, successive values of $Y(t)$ are well correlated. On the other hand, successive values of $Y(t)$ over time intervals that are large with respect to T are nearly uncorrelated. Therefore, in this second limit, we have

$$\int_0^t R_y(s)ds \backsim T \Rightarrow \langle X^2\rangle \backsimeq 2\langle Y^2\rangle T \times t \tag{2.3.36}$$

That is the mean square of $X(t)$ varies linearly with the time interval, here t. That is the 'diffusive regime' (see Appendix A).

As we are concerned in modelling the instantaneous *trajectories*, if we assume that the trajectories of $X(t)$ are continuous, the previous result suggests that, in the range $t \gg T$, $X(t)$ can be seen as a **Wiener process**. This behaviour is obtained by first introducing T and then making t or Δt large enough. Now imagine to reverse the reasoning, so that we introduce a time step $\Delta t \backsim dt$ representing the time interval over which we observe the process $X(t)$, and we assume that the integral time scale of $Y(t)$, T, is very small with respect to dt. Thus, $Y(t)$ is a fast and rapidly changing variable. Actually, we would like to take directly the limit $T \to 0$, but if we take that limit, Eq. 2.3.36 shows that the effect of the fluctuations of $Y(t)$ vanishes completely. Consequently, to retain a finite limit when $T \to 0$, we are forced to consider that $\langle Y^2\rangle$ becomes arbitrary large such that

$$\begin{cases} \langle Y^2\rangle \to +\infty \\ T \to 0 \end{cases} \quad with \;\; \langle Y^2\rangle T \to D \tag{2.3.37}$$

where D is a finite constant. In that case the modelling step consists in replacing the differentiable process $Y(t)$ by a white noise and writing that $X(t)$ becomes a diffusion process defined by the

SDE

$$dX(t) = \sqrt{2D}dW(t) \qquad with \quad D = \lim_{T\to 0}\langle Y^2\rangle T \tag{2.3.38}$$

By making this step, $X(t)$ becomes a Markov process since the memory of $Y(t)$ becomes infinitesimally small. The significance of this modelling step can be further clarified by writing the consequences in the pdf equation. If $p(t, y)$ is the PDF associated to the process $X(t)$, we have

$$\frac{\partial p}{\partial t} = -\frac{\partial\langle Y(t)|X(t) = y\rangle}{\partial y} \quad T \to 0 \quad \frac{\partial p}{\partial t} = D\frac{\partial^2 p}{\partial y^2} \tag{2.3.39}$$

The guiding principle is then to retain only the slow modes or variables in the state vector used to build the model and to 'eliminate' the fast ones. The latter modes are eliminated by expressing them as functions of the slow ones. Of course, this procedure will be successful if there exist a clear separation of scales between the integral time scales of the slow modes and of the fast ones. A typical example of this reasoning is the historical case of a Brownian particle. For this case a model has been proposed by Langevin in which he retained in the state vector the position and the velocity of the particle, while the acceleration was to be modelled. The stochastic process $U(t)$ generated by the Langevin equation is called the Ornstein-Uhlenbeck process.

2.3.6 Langevin equations

The model proposed by Langevin (1908) for a Brownian particle (see Minier and Peirano [16]) is:

$$\begin{cases} \frac{dX(t)}{dt} = U(t) \\ \frac{dU(t)}{dt} = A(t) \end{cases} \rightarrow \begin{cases} dX(t) = U(t)dt \\ dU(t) = -\frac{U(t)}{T}dt + \sqrt{\frac{2\sigma^2}{T}}dW(t) \end{cases} \tag{2.3.40}$$

and the corresponding PDF equation is the Fokker-Planck equation (see Appendix A)

$$\frac{\partial f_1}{\partial t} = \frac{\partial}{\partial V}\Big[\frac{1}{T}Vf_1\Big] + \frac{\sigma^2}{T}\frac{\partial^2[f_1]}{\partial V^2} \tag{2.3.41}$$

where with f_1 we indicate the Lagrangian conditional PDF $f_1(V; t|V_1, t_1)$. Following the modelling principle discussed above, in practice we are assuming that the particle acceleration is a 'fast mode' (its integral time scale is smaller than a reference time scale, which may be the velocity integral time scale) and thus we are modelling it with a *Gaussian process*. Thanks to the help of characteristics functions, we can find the solution to this equation, which is the normal

distribution

$$f_1(V;t|V_1,t_1) = \mathcal{N}[V_1 e^{-(t-t_1)/T},\ \ \sigma^2(1-e^{-2(t-t_1)/T})] \tag{2.3.42}$$

If now we regard to the autocovariance of successive particle velocities, assuming that the process is statistically stationary, it is given by

$$\begin{aligned} R(s) = \langle U(t_1+s)U(t_1)\rangle = \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} V V_1 f_1(V;t_1+s|V_1,t_1)\, dV_1 dV = \\ \int_{-\infty}^{+\infty} V_1 \Big[\int_{-\infty}^{+\infty} V f_1(V;t_1+s|V_1,t_1)\, dV\Big] dV_1 = \\ \int_{-\infty}^{+\infty} V_1 \langle U(t_1+s)|U(t_1)\rangle f(V_1;t_1)\, dV_1 = \\ \langle\langle U(t_1+s)|U(t_1)\rangle U(t_1)\rangle = \\ \langle U(t_1)^2\rangle e^{-\frac{s}{T}} = \sigma^2 e^{-\frac{s}{T}} \end{aligned} \tag{2.3.43}$$

where in the last two passages we used the results found in Eq. 2.3.42. In normalized form we have the autocorrelation function

$$\rho(s) = e^{-|s|/T} \tag{2.3.44}$$

By the definition of the integral time scale as the integral of the autocorrelation function we can easily find out that the reference time T in the Langevin model is exactly the integral time scale of the velocity.

Even if the Langevin model was originally proposed for modelling Brownian motion, it has been shown that it may provide a good model also for the velocity of a fluid particle in turbulence.

Homogeneous turbulence

Considering the simplest case of homogeneous isotropic turbulence, the mean velocity is zero, and the values of k and ϵ are constant. In these circumstances all fluid particles are statistically identical, and their component of velocity are also statistically identical. It is sufficient therefore to consider one component of the fluid particle velocity, which is denoted by $U^+(t)$. The model for $U^+(t)$ given by the Langevin equation is denoted by $U^*(t)$.

As we have seen before for the case of Brownian motion, the Langevin equation is the stochastic differential equation

$$dU^*(t) = -U^*(t)\frac{dt}{T_L} + \Big(\frac{2\sigma^2}{T_L}\Big)^{1/2} dW(t) \tag{2.3.45}$$

In Eq. 2.3.45 the deterministic drift term $(-U^* dt/T_L)$ causes the velocity to relax toward zero on the time scale T_L, whereas the diffusion term adds a zero-mean random increment of standard deviation $\sigma\sqrt{2dt/T_L}$.
The PDF $f_L^*(V;t)$ of $U^*(t)$ evolves by the *Fokker-Planck* equation

$$\frac{\partial f_L^*}{\partial t} = \frac{1}{T_L}\frac{\partial}{\partial V}(V f_L^*) + \frac{\sigma^2}{T_L}\frac{\partial^2 f_L^*}{\partial V^2} \tag{2.3.46}$$

The Langevin equation is correct in yielding a Gaussian PDF of velocity as it has been shown from experiments and DNS. The specification

$$\sigma^2 = \frac{2}{3}k \tag{2.3.47}$$

produces the correct velocity variance. To fully specify the constants appearing in the Langevin equation we need to resort to the Lagrangian structure function. Considering a high-Reynolds-number turbulence in which there is a large separation between the integral timescale T_L and the Kolmogorov timescale τ_η, we examine $U^+(t)$ through the Lagrangian structure function on inertial-range timescales s, $T_L \gg s \gg \tau_\eta$

$$D_L(s) = \langle [U^+(t+s) - U^+(t)]^2 \rangle \tag{2.3.48}$$

The Kolmogorov hypotheses predict

$$D_L(s) = \mathcal{C}_0 \epsilon s, \qquad for \quad T_L \gg s \gg \tau_\eta \tag{2.3.49}$$

whereas the Langevin equation yields

$$D_L^*(s) = \langle [U^*(t+s) - U^*(t)]^2 \rangle = \frac{2\sigma^2}{T_L} s \tag{2.3.50}$$

Thus the Langevin equation is consistent with the Kolmogorov hypotheses in yielding a linear dependence of D_L on s in the inertial range. So, thanks to this last result and to the specification obtained in Eq. 2.3.47, we can express the Langevin equation in terms of k and ϵ in place of σ^2 and T_L

$$\frac{2\sigma^2}{T_L} = C_0 \epsilon \tag{2.3.51}$$

so that the time scale is given by

$$T_L^{-1} = \frac{C_0\epsilon}{2\sigma^2} = \frac{3}{4}C_0\frac{\epsilon}{k} \tag{2.3.52}$$

where C_0 is a model constant and must not be confused with $\mathcal{C}_0$ which is instead the Kolmogorov constant. With the coefficient re-expressed in this way, the Langevin equation becomes

$$dU^*(t) = -\frac{3}{4}C_0\frac{\epsilon}{k}U^*(t)dt + (C_0\epsilon)^{1/2}dW(t) \tag{2.3.53}$$

The Langevin equation is quantitatively consistent with the Kolmogorov hypotheses if the model coefficient C_0 is taken to be the Kolmogorov constant $\mathcal{C}_0$. In that case $D_L(s)$ and $D_L^*(s)$ are identical (in the inertial range).

Inhomogeneous turbulence

The extension of the Langevin equation (Eq. 2.3.53) to inhomogeneous turbulent flows leads to the *simplified Langevin model* (SLM), from which can then be obtained the *generalized Langevin model* (GLM). The subject of these Langevin equations is the velocity $\mathbf{U}^*(t)$ of a particle with position $\mathbf{X}^*(t)$. The particle models the behaviour of fluid particles and consequently it moves with its own velocity

$$\frac{d\mathbf{X}^*(t)}{dt} = \mathbf{U}^*(t) \tag{2.3.54}$$

The simplified Langevin model is written as the stochastic differential equation

$$dU_i^*(t) = -\frac{1}{\rho}\frac{\partial\langle p\rangle}{\partial x_i}dt - (\frac{1}{2}+\frac{3}{4}C_0)\frac{\epsilon}{k}(U_i^*(t)-\langle U_i\rangle)dt + (C_0\epsilon)^{1/2}dW_i(t) \tag{2.3.55}$$

where $\mathbf{W}(t)$ is a vector-valued Wiener process, and the coefficients ($\partial\langle p\rangle/\partial x_i$, k, ϵ, and $\langle U_i\rangle$) are evaluated at the particle's location $\mathbf{X}^*(t)$.

The differences from the scalar Langevin equation (Eq. 2.3.53) are the two drift terms (mean pressure and mean velocity). The drift term in the mean pressure gradient is an exact term from the Navier-Stokes equations. In the second drift term the particle velocity $\mathbf{U}^*(t)$ relaxes toward the local Eulerian mean (in homogeneous isotropic turbulence the mean velocity is zero). The additional 1/2 in the coefficient $(\frac{1}{2}+\frac{3}{4}C_0)$ is necessary to have the correct energy-dissipation rate ϵ. These modifications are the minimum necessary for consistency with the mean momentum

and kinetic energy equations. We can then easily generalize the SLM to obtain the *generalized Langevin model* (GLM)

$$dU_i^*(t) = -\frac{1}{\rho}\frac{\partial \langle p \rangle}{\partial x_i}dt - G_{ij}(U_j^*(t) - \langle U_j \rangle)dt + (C_0\epsilon)^{1/2}dW_i(t) \tag{2.3.56}$$

where the coefficient $G_{ij}(\boldsymbol{x}, t)$ depends on the local values of $\langle u_i u_j \rangle$, ϵ, and $\partial \langle U_i \rangle / \partial x_j$.
In the generalized Langevin model, $\mathbf{X}^*(t)$ and $\mathbf{U}^*(t)$ model the fluid-particle properties and hence their joint PDF f_L^* is a model for f_L. The evolution equation for f_L^* is simply the Fokker-Planck equation as seen above

$$\frac{\partial f_L^*}{\partial t} = -V_i\frac{\partial f_L^*}{\partial x_i} + \frac{1}{\rho}\frac{\partial \langle p \rangle}{\partial x_i}\frac{\partial f_L^*}{\partial V_i} - G_{ij}\frac{\partial}{\partial V_i}[f_L^*(V_j - \langle U_j \rangle)] + \frac{1}{2}C_0\epsilon\frac{\partial^2 f_L^*}{\partial V_i \partial V_i} \tag{2.3.57}$$

where all the coefficients are evaluated at $(\boldsymbol{x}, t)$. On the basis of the relationship between Lagrangian and Eulerian PDFs, the model Eulerian PDF corresponding to the generalized Langevin model is

$$f^*(\mathbf{V}; \boldsymbol{x}, t) = \int f_L^*(\mathbf{V}, \boldsymbol{x}; t|\mathbf{Y})d\mathbf{Y} \tag{2.3.58}$$

The evolution equation for f^* is readily obtained by integrating the equation for f_L^* over all $\mathbf{Y}$. Since this equations contains no dependence on $\mathbf{Y}$, the result is simply

$$\frac{\partial f^*}{\partial t} = -V_i\frac{\partial f^*}{\partial x_i} + \frac{1}{\rho}\frac{\partial \langle p \rangle}{\partial x_i}\frac{\partial f^*}{\partial V_i} - G_{ij}\frac{\partial}{\partial V_i}[f^*(V_j - \langle U_j \rangle)] + \frac{1}{2}C_0\epsilon\frac{\partial^2 f^*}{\partial V_i \partial V_i} \tag{2.3.59}$$

This equation is precisely the model Eulerian PDF equation examined in Sec. 2.3.2.
For computational reasons, the equation to be solved is not the evolution equation of the PDF, but is the Langevin equation for a very large number of particles (Monte Carlo method). Then from the solutions, by evaluating the means we can find back all RANS quantities. This method can be built in a stand-alone code in which at each time step we will have also to evaluate the mean velocity and the mean pressure to put in the Langevin equations. More frequently a hybrid approach is adopted, where the PDF solver is coupled with a RANS solver which directly gives to the PDF module the mean quantities. Since for the fluid part we are using a LES solver, as said previously, we are more interested in another kind of approach, which is very similar to the PDF one, but instead of treating the probability density function, it treats the so called *Velocity Filtered Density Function* (VFDF).

2.4 Velocity Filtered Density Function

This approach is very similar to the PDF one and it's for this reason that the PDF methods have been introduced nevertheless they are not used. This class of method is analysed in more details by Colucci *et al.*[13], Gicquel *et al.*[14], Sheikhi *et al.*[15]. The *velocity filtered density function* (VFDF), denoted by P_L, is formally defined as

$$P_L(\mathbf{V};\boldsymbol{x},t) = \int_{-\infty}^{+\infty} \varrho[\mathbf{V},\mathbf{U}(\boldsymbol{x}',t)]G(\boldsymbol{x}'-\boldsymbol{x})d\boldsymbol{x}' \tag{2.4.1}$$

$$\varrho[\mathbf{V},\mathbf{U}(\boldsymbol{x},t)] = \delta[\mathbf{V}-\mathbf{U}(\boldsymbol{x},t)] = \prod_{i=1}^{3}\delta[V_i - U_i(\boldsymbol{x},t)] \tag{2.4.2}$$

where δ denotes the delta function, $\mathbf{V}$ is the velocity state vector and G is the filter function. The term $\varrho[\mathbf{V},\mathbf{U}(\boldsymbol{x},t)]$ is the 'fine-grained' density, analogously to the one used in the PDF formalism, and Eq. 2.4.1 defines the VFDF as the spatially filtered value of the fine-grained density. With the condition of a positive filter kernel, P_L has all the properties of the PDF. It is now useful to define the "conditional filtered value" of the variable $Q(\boldsymbol{x},t)$ by

$$\langle Q(\boldsymbol{x},t)|\mathbf{U}(\boldsymbol{x},t)=\mathbf{V}\rangle_L = \langle Q|\mathbf{V}\rangle_L = \frac{\int_{-\infty}^{+\infty} Q(\boldsymbol{x}',t)\varrho[\mathbf{V},\mathbf{U}(\boldsymbol{x}',t)]G(\boldsymbol{x}'-\boldsymbol{x})d\boldsymbol{x}'}{P_L(\mathbf{V};\boldsymbol{x},t)} \tag{2.4.3}$$

where $\langle\alpha|\beta\rangle_L$ denotes the filtered value of α conditioned on β. Eq. 2.4.3 implies

- for $Q(\boldsymbol{x},t)=c, \quad \langle Q(\boldsymbol{x},t)|\mathbf{V}\rangle_L = c$
- for $Q(\boldsymbol{x},t)=\hat{Q}(\mathbf{U}(\boldsymbol{x},t)), \quad \langle Q(\boldsymbol{x},t)|\mathbf{V}\rangle_L = \hat{Q}(\mathbf{V})$
- $$\langle Q(\boldsymbol{x},t)\rangle_L = \int_{-\infty}^{+\infty} \langle Q(\boldsymbol{x},t)|\mathbf{V}\rangle_L P_L(\mathbf{V};\boldsymbol{x},t)d\mathbf{V} \tag{2.4.4}$$

where c is a constant, and $Q(\boldsymbol{x},t)=\hat{Q}(\mathbf{U}(\boldsymbol{x},t))$ denotes the case where the variable Q is completely described by the variable $\mathbf{U}(\boldsymbol{x},t)$. From these properties it follows that the filtered value of any function of the velocity variable is obtained by integration over the velocity space

$$\langle Q(\boldsymbol{x},t)\rangle_L = \int_{-\infty}^{+\infty} \hat{Q}(\mathbf{V})P_L(\mathbf{V};\boldsymbol{x},t)d\mathbf{V} \tag{2.4.5}$$

where $\langle Q(\boldsymbol{x},t)\rangle_L$ follows exactly the definition given in the LES approach (Sec. 2.2). This result is very important as it shows the close similarity between the averaging operator and the filtering

operator if we think to P_L as to a density function. So we now follow exactly the same passages done in the PDF approach. First of all we evaluate the VFDF transport equation. We simply use the result obtained for the fine-grained PDF (see Sec. 2.3.2) and multiplying Eq. 2.3.12 by $G(\boldsymbol{x}' - \boldsymbol{x})$ and integrating over $d\boldsymbol{x}'$, we obtain

$$\frac{\partial P_L(\mathbf{V};\boldsymbol{x},t)}{\partial t} + V_i \frac{\partial P_L(\mathbf{V};\boldsymbol{x},t)}{\partial x_i} = -\frac{\partial}{\partial V_i}\Big[\Big\langle \frac{DU_i}{Dt}\Big|\mathbf{V}\Big\rangle_L P_L(\mathbf{V};\boldsymbol{x},t)\Big] \tag{2.4.6}$$

where to obtain the term on RHS we used also Eq. 2.4.3. From now on we can follow exactly the same procedure we adopted in the PDF approach. So we substitute the Navier-Stokes equation into Eq. 2.4.6 and we decompose into filtered and unresolved values, so that we obtain

$$\begin{aligned}\frac{DP_L}{Dt} = -\frac{\partial}{\partial x_k}[(V_k - \langle U_k\rangle_L)P_L] + \frac{1}{\rho}\frac{\partial\langle p\rangle_L}{\partial x_i}\frac{\partial P_L}{\partial V_i} - \nu\nabla^2\langle U_i\rangle_L\frac{\partial P_L}{\partial V_i} \\ + \frac{\partial}{\partial V_i}\Big[\Big(\Big\langle\frac{1}{\rho}\frac{\partial p}{\partial x_i}\Big|\mathbf{V}\Big\rangle_L - \frac{1}{\rho}\frac{\partial\langle p\rangle_L}{\partial x_i}\Big)P_L\Big] - \\ \frac{\partial}{\partial V_i}\Big[\Big(\Big\langle\nu\nabla^2 U_i\Big|\mathbf{V}\Big\rangle_L - \nu\nabla^2\langle U_i\rangle_L\Big)P_L\Big]\end{aligned} \tag{2.4.7}$$

where $D/Dt = \partial/\partial t + \langle U_k\rangle_L(\partial/\partial x_k)$ denotes the 'filtered material derivative'. The first line corresponds to the effects of the resolved scale (closed terms), while the last two lines correspond to the effects of the unresolved values (unclosed terms). In effect, in our model we didn't care of large scale dissipation, as in the PDF model we didn't care of the dissipation of the mean field. This for two reasons: because we aim to elaborate a simply model and because that term is not so relevant if compared to the changes that can be obtained varying even a little the model form. So next step is to replace the unclosed terms with a model. Thus, we can use the same model used in the PDF approach, simply replacing the averaged quantities by the filtered quantities.

$$\begin{aligned}\frac{DP_L}{Dt} = -\frac{\partial}{\partial x_k}[(V_k - \langle U_k\rangle_L)P_L] + \frac{1}{\rho}\frac{\partial\langle p\rangle_L}{\partial x_i}\frac{\partial P_L}{\partial V_i} \\ + \frac{\partial}{\partial V_i}[G_{ij}(V_j - \langle U_j\rangle_L)P_L] + \frac{1}{2}C_0\epsilon_r\frac{\partial^2 P_L}{\partial V_i\partial V_i}\end{aligned} \tag{2.4.8}$$

Multiplying by $\mathbf{V}$ and integrating over the sample space we can obtain the momentum filtered equations and verify that they are exactly the same that we obtain by filtering the Navier-Stokes equations unless the viscous term. The same result of consistency is obtained for the SGS stresses, if we multiply the above equation by the subgrid velocity and integrate.

So, keeping on treating the velocity filtered density function as a PDF, we can transfer all the

reasoning made above on the duality between the PDF transport equation approach and the Lagrangian one, modelling the particles transport equation as a stochastic diffusion process. Hence, the SDEs which represent the modelled VFDF in the Lagrangian sense are

$$dX_i^*(t) = U_i^*(t) \tag{2.4.9}$$

$$dU_i^*(t) = -\frac{1}{\rho}\frac{\partial \langle p \rangle_L}{\partial x_i}dt + G_{ij}(U_j^*(t) - \langle U_j \rangle_L)dt + \sqrt{C_0 \epsilon_r}\, dW_i(t) \tag{2.4.10}$$

So, if we replace G_{ij} by the simplified Langevin model, we obtain our reference equation for the fluid modelling

$$dU_i^*(t) = -\frac{1}{\rho}\frac{\partial \langle p \rangle_L}{\partial x_i}dt - (\frac{1}{2} + \frac{3}{4}C_0)\frac{\epsilon_r}{k_{SGS}}(U_i^*(t) - \langle U_i \rangle_L)dt + \sqrt{C_0 \epsilon_r}\, dW_i(t) \tag{2.4.11}$$

where ϵ_r and k_{SGS} are the residual dissipation and energy.

Chapter 3

Particle model

3.1 Extension from the fluid model to the particle model

Following what we said in the introduction, our aim is actually to elaborate a stochastic model for particle tracking in a LES contest. While for the fluid phase we will adopt a traditional LES approach (it will be discussed more in detail in the next chapter), for the dispersed phase we are using a VFDF methodology, that can be simply seen as an extension from the fluid case. We now briefly recall the equations governing particle motion, considering only drag and gravity forces as we are concerned with heavy small particle

$$\begin{cases} \dfrac{d\mathbf{x}_p(t)}{dt} = \mathbf{U}_p(t) \\ \dfrac{d\mathbf{U}_p(t)}{dt} = \dfrac{1}{\tau_p}(\mathbf{U}_s(t) - \mathbf{U}_p(t)) + \mathbf{g} \end{cases} \tag{3.1.1}$$

In these equations, $\mathbf{U}_s(t) = \mathbf{U}(t, \mathbf{x}_p(t))$ is the fluid velocity "seen", i.e. the fluid velocity sampled along the particle trajectory $\mathbf{x}_p(t)$, where $\mathbf{U}(t, \mathbf{x})$ is the local instantaneous (Eulerian) fluid velocity field. The particle relaxation time, τ_p, is defined as

$$\tau_p = \frac{\rho_p}{\rho_f} \frac{4d_p}{3C_D|\mathbf{U}_r|}, \tag{3.1.2}$$

where the local instantaneous relative velocity is $\mathbf{U}_r(t) = \mathbf{U}_p(t) - \mathbf{U}_s(t)$. The drag coefficient, C_D, is a non-linear function of the particle-based Reynolds number, $Re_p = d_p|\mathbf{U}_r|/\nu_f$, which means that C_D is a non-linear function of the particle diameter, d_p.

We remark that since we aim to resolve exactly this set of equations, we are dealing with instantaneous fluid velocities, $\mathbf{U}(t, \mathbf{x}_p(t))$, which would be known only if we would adopt a DNS approach to resolve the fluid. Instead we only know LES velocities, $\langle \mathbf{U} \rangle_L(t, \mathbf{x}_p(t))$, so a modelling step is necessary and a statistical approach is adopted. In practice, the particles-VFDF approach has the form of a particle stochastic method where the velocity of the fluid seen, $\mathbf{U}_s(t)$, is modelled as a stochastic diffusion process, i.e. the dynamics of the particles are calculated from SDEs, the so-called Langevin equations (see Chapter 2). In polydispersed two-phase flows, a particle point of view seems rather natural, given the physics considered. Yet, the particles which are to be simulated represent samples of the FDF and should not be confused with real particles. Within the VFDF formalism, this particle point of view is helpful to build the theoretical model and, at the same time, represents directly a discrete formulation of the model.

We restrict ourselves to situations where only particle dispersion is the important issue. The

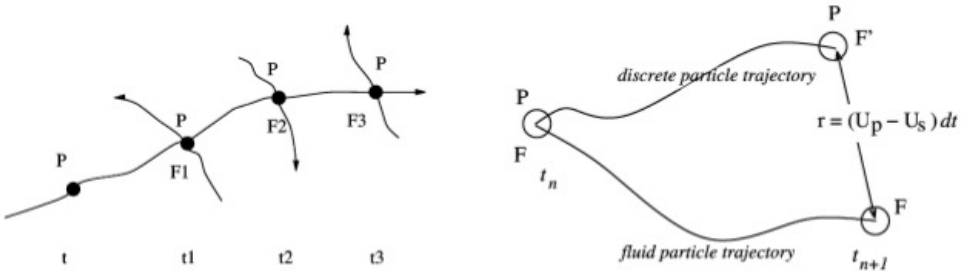


Figure 3.1: Crossing trajectory effect

modelling problem is sketched in Fig. 3.1. The problem is more complicated than pure diffusion models. Indeed, compared to a fluid particle, the determination of the fluid velocity seen is further compounded by particle inertia (τ_p) and the effect of an external force field (gravity in our case g). Both effects induce a separation of the fluid element and of the discrete particle which are located near the same point at the beginning of the time interval. This is represented in Fig. 3.1 between two discrete time steps t_n and t_{n+1}. In the absence of gravity (or other external force) and for small particle inertia, $\tau_p \to 0$, the separation effect disappears and, in that limit, the modelling issue is to represent the successive velocities of a fluid particle, for which the stochastic models developed in Chapter 2 can be applied. Two main approaches can be found in the literature:

- approaches based on paths (trajectories). A two-step construction is considered: a La-

grangian step and an Eulerian step. The Lagrangian step corresponds to the trajectory, over a time interval dt, of a fluid particle located at time t in the vicinity of the discrete particle (this step is directly given by Eq. 2.4.11). The Eulerian step corresponds to a spatial correction which gives, from the location of the fluid particle at $t + dt$, the fluid velocity seen by the discrete particle at time $t + dt$. This modelling point of view has two major drawbacks: it leads to an artificial decrease of the integral time scale of $\mathbf{U}_s(t)$ (denoted $T^*_{L,i}$ in the present paper) and there is no clear separation between the effects of τ_p and $\mathbf{g}$.

- approaches based on the physical effects. A two-step construction is also considered by decoupling the two physical mechanisms: the first step corresponds to the effects of τ_p in the absence of external forces (in that case $T^*_{L,i}$ varies between two limit values, T_E -the integral Eulerian time scale- when $\tau_p \to +\infty$ and T_L when $\tau_p \to 0$). The second step corresponds to the effects of gravity alone which induce a mean drift and result in a decorrelation of $\mathbf{U}_s(t)$ with respect to $\mathbf{U}_f(t)$. This effect is called the crossing trajectory effect (CTE) and is related to the filtered relative velocity $\langle \mathbf{U}_r \rangle_L = \langle \mathbf{U}_p - \mathbf{U}_s \rangle_L$.

In the present work, the derivation of a model for $\mathbf{U}_s(t)$ is carried out by resorting to an approach based on the physical effects where the influence of gravity is neglected, following the same approach adopted by Minier & Peirano [16], and extending it to the LES case.
Assuming, for the sake of simplicity, that the mean drift (the filtered relative velocity $\langle \mathbf{U}_r \rangle_L$) is aligned with one of the coordinate axes, it can be shown that a possible model for the increments of the fluid velocity seen is

$$\begin{aligned} dU_{s,i}(t) = &-\frac{1}{\rho_f}\frac{\partial \langle p \rangle_L}{\partial x_i}\,dt + \left(\langle U_{p,j} \rangle_L - \langle U_j \rangle_L\right)\frac{\partial \langle U_i \rangle_L}{\partial x_j}\,dt \\ &- \frac{1}{T^*_{L,i}}\left(U_{s,i} - \langle U_i \rangle_L\right)\,dt + \sqrt{\epsilon_r\left(C_0 b_i + \frac{2}{3}(b_i - 1)\right)}\,dW_i(t). \end{aligned} \tag{3.1.3}$$

The CTE has been modelled by changing the time scale, compared to the fluid case, in the drift term (third term on the RHS) and by adding a mean drift term (second term on the RHS). The time scale is modified according to Csanady's analysis

$$T^*_{L,i} = T_L\left(1 + \beta_i^2 \frac{|\langle \mathbf{U}_r \rangle_L|^2}{2k_{SGS}/3}\right)^{-1/2}, \tag{3.1.4}$$

where $\beta_1 = \beta$, if axis 1 is aligned with the mean drift, with $\beta = T_L/T_E$, and in the transversal directions (axes labelled 2 and 3) $\beta_i = 2\beta$. In the diffusion matrix, the term b_i is simply $b_i = T_L/T^*_{L,i}$. Equation (3.1.3) has two noteworthy properties:

- it is consistent, by construction, with Eq. 2.4.11 when $\tau_p \to 0$, that is when the discrete particles behave like fluid particles,

- it is a Mac-Kean SDE even though the filtered fields of the fluid are known (they are given by solving LES equations). Indeed, it is necessary to compute the filtered velocity of the particles $\langle \mathbf{U}_p \rangle_L$ to calculate not only the mean drift term (second term on the RHS) but also the integral time scale of $\mathbf{U}_s(t)$, T^*_L ($\langle \mathbf{U}_s \rangle_L$ is also needed for the computation of this time scale).

Moreover, it must be emphasised that Eq. (3.1.3) is a possible choice among others and that the exact form of a Langevin equation for $\mathbf{U}_s(t)$ still remains an open issue. There exists an alternative to Eq. (3.1.3) in the literature where the coefficients are slightly different (the drift vector and the diffusion matrix), the main difference being the form of the mean drift term which is written in terms of instantaneous velocities rather that filtered velocities, i.e. $(U_{p,j} - U_{s,j})(\partial \langle U_i \rangle / \partial x_j)$. This form of the mean drift term does not change the methodology which is presented in the rest of the work, but it modifies the structure of the system of SDEs, i.e. $\mathbf{U}_s(t)$ depends explicitly on the particle velocity, $\mathbf{U}_p(t)$.

The complete set of SDEs which describes the one-point dynamical behaviour of the discrete particles is

$$\begin{cases} dx_{p,i}(t) = U_{p,i}\, dt, \\ dU_{p,i}(t) = \dfrac{U_{s,i} - U_{p,i}}{\tau_p}\, dt + g_i\, dt, \\ dU_{s,i}(t) = A_{s,i}\, dt + B_{s,ij}\, dW_j(t), \end{cases} \tag{3.1.5}$$

where $\mathbf{A}_s$ and $\mathbf{B}_s$ are calculated by resorting to Eq. (3.1.3)-(3.1.4).

3.2 Limit cases

The system of SDEs describing the dynamics of the discrete particles reads (*from now on $B_{s,ij}$ is denoted B_{ij} for the sake of simplicity*)

$$\begin{cases} dx_{p,i}(t) = U_{p,i}\,dt \\ dU_{p,i}(t) = \dfrac{1}{\tau_p}(U_{s,i} - U_{p,i})\,dt + \mathcal{A}_i\,dt \\ dU_{s,i}(t) = -\dfrac{1}{T^*_{L,i}}U_{s,i}\,dt + C_i\,dt + \displaystyle\sum_j B_{ij}\,dW_j(t) \end{cases} \tag{3.2.1}$$

where C_i is a term that includes all mean contributions: the filtered pressure gradient, $-(\partial\langle p\rangle_L/\partial x_i)/\rho_f$, the mean drift term, $(\langle U_{p,j}\rangle_L - \langle U_j\rangle_L)(\partial\langle U_i\rangle_L/\partial x_j)$, and the filtered part of the return-to-equilibrium term, $\langle U_i\rangle_L/T^*_{L,i}$. There are three different time scales describing the dynamics of the discrete particles: dt, the time scale at which the process is observed, $T^*_{L,i}$ the integral time scale of the fluid velocity seen, $\mathbf{U}_s(t)$, and τ_p the particle relaxation time. System (3.2.1) has a physical meaning only in the case where $dt \ll T^*_{L,i}$ and $dt \ll \tau_p$. When these conditions are not satisfied, it is possible to show that, in the *continuous sense* (time and all coefficients are continuous functions which can go to zero), the system converges towards several limit systems.

Case 1: when $\tau_p \to 0$, the particles behave as fluid particles and one has

$$\text{system (3.2.1)} \xrightarrow[\tau_p\to 0]{} \begin{cases} dx_{p,i}(t) = U_{p,i}\,dt \\ U_{p,i}(t) = U_{s,i}(t) \\ dU_{s,i}(t) = -\dfrac{1}{\rho_f}\dfrac{\partial\langle p\rangle_L}{\partial x_i}\,dt - \dfrac{1}{T_L}(U_{s,i} - \langle U_i\rangle_L)\,dt + \\ \qquad\qquad\qquad\qquad \sqrt{C_0\epsilon}\,dW_i(t) \end{cases} \tag{3.2.2}$$

that is, the model is consistent with a known turbulent fluid VFDF model as explained in Chapter 2. This shows that the model is a coherent generalisation of the fluid one, which can be recovered as a limit case.

Case 2: when $T^*_{L,i} \to 0$ and $B_{ij}T^*_{L,i} \to$ cst, the fluid velocity seen becomes a fast variable. It is

then eliminated and one can write

$$\text{system (3.2.1)} \xrightarrow[T^*_{L,i}\to 0]{(B_{ij}T^*_{L,i}\to \text{cst})} \begin{cases} dx_{p,i}(t) = U_{p,i}\,dt \\ dU_{p,i}(t) = \dfrac{1}{\tau_p}(\langle U_i\rangle_L - U_{p,i})\,dt + \mathcal{A}_i\,dt + \\ \qquad\qquad \displaystyle\sum_j \frac{B_{ij}T^*_{L,i}}{\tau_p}\,dW_j(t) \end{cases} \tag{3.2.3}$$

This result can be understood from the model problem of the previous Chapter. $\mathbf{U}_s(t)$ has been eliminated but its influence is left in the diffusion coefficient $B_{ij}T^*_{L,i}/\tau_p$. In this case, the equations are equivalent to a Fokker-Planck model for particles of significant inertia.

Case 3: When τ_p, $T^*_{L,i} \to 0$ and at the same time $B_{ij}T^*_{L,i} \to$ cst , the fluid velocity seen becomes a fast variable and the discrete particles behave as fluid particles. It can be shown that

$$\text{system (3.2.1)} \xrightarrow[\tau_p, T^*_{L,i}\to 0]{(B_{ij}T^*_{L,i}\to \text{cst})} \begin{cases} dx_{p,i}(t) = \langle U_i\rangle_L\,dt + \mathcal{A}_i\,dt + \\ \qquad\qquad \displaystyle\sum_j (B_{ij}T^*_{L,i})\,dW_j(t) \end{cases} \tag{3.2.4}$$

We retrieve a pure diffusive behaviour, that is the equations of *Brownian motion*.

Case 4: at last, when $T^*_{L,i} \to 0$ with no condition on $B_{ij}T^*_{L,i}$, the velocity of the fluid seen is no longer random and the system becomes deterministic. The flow is laminar and it can be proven that

$$\text{system (3.2.1)} \xrightarrow[T^*_{L,i}\to 0]{} \begin{cases} dx_{p,i}(t) = U_{p,i}\,dt \\ dU_{p,i}(t) = \dfrac{1}{\tau_p}(\langle U_i\rangle_L - U_{p,i})\;dt + \mathcal{A}_i\,dt, \\ U_{s,i}(t) = \langle U_i\rangle_L \end{cases} \tag{3.2.5}$$

Limit cases 1 to 3 reflect the multiscale character of the problem. When the timescales go to zero (with a condition on their products with the coefficients of the diffusion matrix), a hierarchy of stochastic differential systems is obtained. Moreover, the elimination of the fast variables (the velocities $\mathbf{U}_p(t)$ and $\mathbf{U}_s(t)$) does not mean that these variables do not (physically) exist anymore: they simply become Gaussian white noise.

The existence of limit systems is a key point in the development of weak numerical schemes to integrate in time the set of SDEs describing the dynamics of the discrete particles, i.e. Eqs. 3.2.1. As we shall see in the next Section, in numerical computations, dt the observation timescale of the process, is the time step. A suitable weak numerical scheme should therefore be consistent with all limit cases since, as we shall see, it is not possible to control the time step to enforce the

conditions necessary for the validity of Eqs. 3.2.1.

Chapter 4

Numerical approach

The LES/VFDF model used in the present work for practical computations has now been given. It consists in a set of PDEs describing the dynamics of filtered fluid quantities and a set of SDEs. In this approach, the numerical solution is obtained by resorting to a hybrid method where the filtered-fluid properties are computed by solving the LES equations with a classical pseudospectral method whereas the local instantaneous properties of the discrete particles are determined by solving the set of SDEs, Eqs. 3.1.5. Therefore, the filtered fluid properties are computed on a mesh whereas the statistics of the discrete phase are calculated from particles moving in the computational domain. In particular, numerical solution of the modelled stochastic equation is obtained by a Lagrangian Monte Carlo procedure in which particles statistics are simulated by an ensemble of N statistically identical Monte Carlo particles. Each of these particles carries informations pertaining to its velocity and position which are updated at each time step through a weak numerical scheme illustrated after. We must underline that these are not physical particles but statistical, which then can be used to recover statistical informations, like concentration and velocity statistics. In practice, the direct connection between statistics derived from the particle system and LES filtered variables is given by the general PDF formalism. By taking statistics from the local number of particles N_x located within a small volume dV_x centered around a given point $\mathbf{x}$, we have that

$$\langle U_{s,i}(\mathbf{x})\rangle_L \simeq \frac{1}{N_x}\sum_{n=1}^{N_x} U_{s,i}^{(n)} \tag{4.0.1}$$

$$\langle \tau_{ij}(\mathbf{x}) \rangle_L \simeq \frac{1}{N_x} \sum_{n=1}^{N_x} (U_{s,i} - \langle U_{s,i}(\mathbf{x}) \rangle_L) \times (U_{s,j} - \langle U_{s,j}(\mathbf{x}) \rangle_L) \tag{4.0.2}$$

and analogously for all other filtered quantities. By averaging particles quantities within a small volume we recover thus LES quantities because we are simulating the underlying FDF and not the PDF (from which we would obtain averaged quantities). In practice we are trading on Eq. 2.4.4 which let us treating filtering as an averaging operator.

As we have just mentioned we adopt a hybrid method which couples the LES solver with Lagrangian particle tracking, so particular attention must be given also to the procedure used for coupling. Let $\{\mathbf{Y}^{[\mathbf{x}]}\}$ stand for the set of the fluid filtered fields at the different mesh points and let $\{\mathbf{Y}^{(N)}\}$ be the fluid filtered fields interpolated at particle locations. Let $\{\mathbf{Z}^{(N)}\}$ denote the set of variables attached to the stochastic particles and $\{\mathbf{Z}^{[\mathbf{x}]}\}$ the set of statistics, defined at cell centres, extracted from $\{\mathbf{Z}^{(N)}\}$. Time is discretised with a constant time step $\Delta t = t_{n+1} - t_n$ and space with a non uniform mesh with cells of constant size in the streamwise and spanwise directions while varying in the wall-normal direction (Chebyshev points).

The first step (operator F) is to solve the PDEs describing the fluid (done by FLOWSB),

$$\{\mathbf{Y}^{[\mathbf{x}]}\}(t_n) \xrightarrow{F} \{\mathbf{Y}^{[\mathbf{x}]}\}(t_{n+1}). \tag{4.0.3}$$

The second step (projection, operator P) consists in calculating filtered fluid properties and filtered particle properties at particle locations,

$$\{\mathbf{Z}^{[\mathbf{x}]}\}(t_n) \text{ and } \{\mathbf{Y}^{[\mathbf{x}]}\}(t_n) \xrightarrow{P} \{\mathbf{Z}^{(N)}\}(t_n) \text{ and } \{\mathbf{Y}^{(N)}\}(t_n), \tag{4.0.4}$$

Then, the stochastic differential system can be integrated in time (operator T),

$$\{\mathbf{Z}^{(N)}\}(t_n) \text{ and } \{\mathbf{Y}^{(N)}\}(t_n) \xrightarrow{T} \{\mathbf{Z}^{(N)}\}(t_{n+1}). \tag{4.0.5}$$

Finally, from the new computed set of variables, at particle locations, new statistical moments are evaluated at cell centres,

$$\{\mathbf{Z}^{(N)}\}(t_{n+1}) \xrightarrow{A} \{\mathbf{Z}^{[\mathbf{x}]}\}(t_{n+1}), \tag{4.0.6}$$

and so on.

4.1 FLOWSB

Attention is now focused in spatial and time discretization of LES equations (operator F). As usually happens in fluid dynamic studies, equation of fluid motion are presented in dimensionless form. As we will solve the equations over a channel, the computational domain sketched in Figure 4.1, has dimensions:

$$L_x = 4\pi h \ , \ L_y = 2\pi h \ , \ L_z = h \tag{4.1.1}$$

where x, y and z coordinates represent the stream-wise, span-wise and wall normal directions. We can get the dimensionless form of the equations by using the half-height h as characteristic

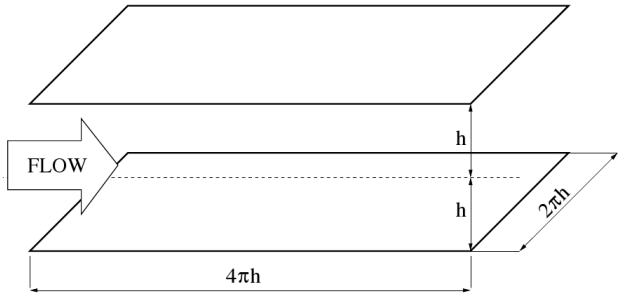


Figure 4.1: Reference domain for LES channel flow.

length scale and the shear velocity:

$$u_\tau = \sqrt{\frac{\tau_w}{\rho}} \tag{4.1.2}$$

where, τ_w is the mean shear wall, as velocity reference. In this case, we have introduced the outer-scaling units (indicated by '-' apex) and variables are put in dimensionless form as follow:

$$\mathbf{x}^- = \frac{\mathbf{x}}{h} \ , \ \mathbf{u}^- = \frac{\mathbf{u}}{u_\tau} \ , \ p^- = \frac{p}{\rho u_\tau^2} \ , \ t^- = \frac{t}{h/u_\tau} \ , \tag{4.1.3}$$

By simple manipulation, omitting for simplicity the filtering symbol $\langle \, \rangle_L$, the LES equations (Eq. 2.2.3) become:

$$\frac{\partial u_j^-}{\partial x_j^-} = 0 \tag{4.1.4}$$

$$\frac{\partial u_i^-}{\partial t^-} + \frac{\partial (u_i^- u_j^-)}{\partial x_j^-} = -\frac{\partial P^-}{\partial x_i^-} + \frac{1}{Re_\tau} \frac{\partial^2 u_i^-}{\partial x_i^- \partial x_j^-} - \frac{\partial \tau_{ij}^-}{\partial x_j^-} \tag{4.1.5}$$

Where we have defined the shear Reynolds number Re_τ as follow:

$$Re_\tau = \frac{\rho u_\tau h}{\mu} \tag{4.1.6}$$

In particular, Navier-Stokes equations can be rewritten compactly, neglecting the apex notation, if we split the pressure term on average and fluctuating part so that $p = \langle p \rangle + p'$:

$$\frac{\partial u_i}{\partial t} = -\frac{\partial p}{\partial x_i} + \frac{1}{Re_\tau}\frac{\partial^2 u_i}{\partial x_i \partial x_j} + S_i \tag{4.1.7}$$

where with S_i (source term) we collect the non-linear transport term, the SGS stress term and the mean pressure gradient that drives the flow (imposed -1 in outer units). Thus, the three components of source term are:

$$\begin{aligned} S_1 = -\frac{\partial(u_1 u_j)}{\partial x_j} - \frac{\partial \tau_{1j}}{\partial x_j} + 1 \ , \ S_2 = -\frac{\partial(u_2 u_j)}{\partial x_j} - \frac{\partial \tau_{2j}}{\partial x_j}, \\ S_3 = -\frac{\partial(u_3 u_j)}{\partial x_j} - \frac{\partial \tau_{3j}}{\partial x_j} \end{aligned} \tag{4.1.8}$$

4.1.1 Rephrased equations

In the case of numerical simulation , it is convenient to rewrite the LES equations without taking explicitly in count the pressure gradient term.
This can be done by taking the curl of equation (4.1.5) which, on vectorial notation, gives:

$$\nabla \times \frac{\partial \mathbf{u}}{\partial t} = -\nabla \times \nabla p + \frac{1}{Re_\tau}\nabla \times \nabla^2 \mathbf{u} + \nabla \times \mathbf{S} \tag{4.1.9}$$

Reminding the definition of vorticity $\boldsymbol{\omega} = \nabla \times \mathbf{u}$ and that the curl of the gradient of any scalar field is always the zero vector, we get the vorticity transport equation:

$$\frac{\partial \boldsymbol{\omega}}{\partial t} = \nabla \times \mathbf{S} + \frac{1}{Re_\tau}\nabla^2 \boldsymbol{\omega} \tag{4.1.10}$$

If we take the curl of vorticity transport equation, we use the continuity equation ($\nabla \mathbf{u} = 0$) and the vectorial identity $\nabla \times (\nabla \times \mathbf{A}) = \nabla(\nabla \cdot \mathbf{A}) - \nabla^2 \mathbf{A}$ we will get the following fourth order equation:

$$\frac{\nabla^2 \mathbf{u}}{\partial t} = \nabla^2 \mathbf{S} - \nabla(\nabla \cdot \mathbf{S}) + \frac{1}{Re_\tau}\nabla^4 \mathbf{u} \tag{4.1.11}$$

Using the algorithm developed by Kim J., Moin P., Moser R. (1987), to solve the channel flow equations we have to write the following system equations:

$$\begin{cases} w_3, u_3 \leftarrow \begin{cases} \frac{w_3}{\partial t} = \frac{S_2}{\partial x_1} - \frac{S_1}{\partial x_2} + \frac{1}{Re_\tau}\nabla^2 w_3 \\ \frac{(\nabla^2 u_3)}{\partial t} = \nabla^2 S_3 - \frac{\partial}{\partial x_3}\left(\frac{\partial S_j}{\partial x_j}\right) + \frac{1}{Re_\tau}\nabla^4 u_3 \end{cases} \\ \quad \downarrow \\ \begin{cases} \frac{u_1}{\partial x_1} + \frac{u_2}{\partial x_2} = \frac{u_3}{\partial x_3} \\ \frac{u_2}{\partial x_1} - \frac{u_1}{\partial x_2} = w_3 \end{cases} \rightarrow u_1, u_2 \end{cases} \tag{4.1.12}$$

where the first equation and the second equation are the (4.1.10) and the (4.1.11) rewritten for the third component respectively while the third and the forth equation are used to compute the first and the second velocity components starting from continuity and vorticity definition. The system is solved considering no-slip boundary condition at the walls and periodic boundary conditions along x and y directions.

4.1.2 Channel flow equations on spectral space

To solve the system of equations (4.1.12) we use the FLOWSB code own by University of Udine. FLOWSB is a pseudo-spectral Fortran code thus the equations seen in the previous section are solved on the spectral space rather than the physical one. All quantities, starting from velocity field, are decomposed in a Fourier-Chebyshev approximation, where the Fourier modes are used to represent the homogeneous periodic directions (stream-wise and span-wise), while the Chebyshev modes are used for the wall normal direction. In this case the Fourier-Chebyshev representation of the generic component of velocity field permits us to rewrite:

$$u_i(\mathbf{x}, t) = \sum_{n_z=0}^{N_z} \sum_{n_x=1-\frac{1}{2}N_x}^{\frac{1}{2}N_x} \sum_{n_y=1-\frac{1}{2}N_y}^{\frac{1}{2}N_y} \hat{u}_i(k_x, k_y, n_z, t) T_{n_z}(z) e^{i(k_x x + k_y y)} \tag{4.1.13}$$

where:

$$T_{n_z}(z) = \cos(n_z \arccos(x)), \quad n_z = 0, 1, ...N_z \quad z \in [-1, 1] \tag{4.1.14}$$

are the N_z Chebyshev polynomials.

By using this kind of representation, the N_z modes correspond to a non uniform grid spacing on

physical space where:

$$z = \cos(n_z \pi N_z), \quad z \in [-1, 1] \tag{4.1.15}$$

4.1.3 LES solver

By using the (4.1.13) transformation the system of equations 4.1.12 in physical space can be rewritten, in Fourier spectral space, as function of the coefficients. Thus the second equation, for example, will become:

$$\begin{aligned}\frac{\partial}{\partial t}\left[\frac{\partial}{\partial z} - k_x^2 - k_y^2\right]\hat{u}_z = \frac{\partial}{\partial t}\left[\frac{\partial}{\partial z} - k_x^2 - k_y^2\right]\hat{S}_z \\ -\frac{\partial}{\partial z}\left[ik_x\hat{S}_x + ik_y\hat{S}_y + \frac{\partial \hat{S}_z}{\partial z}\right] \\ +\frac{1}{Re_\tau}\frac{\partial}{\partial t}\left[\frac{\partial}{\partial z} - k_x^2 - k_y^2\right]^2 \hat{u}_z\end{aligned} \tag{4.1.16}$$

All the others can also be transformed on Fourier space and rewritten similarly.
The first two equation of the transformed system are solved for the coefficients of the third component of vorticity and velocity. Then they are used to solve the last two equation from which we extract the coefficients of the other two components of the velocity field.
Time advancement of equation (4.1.16) is done by two-level explicit Adams-Bashfort scheme for the convective terms and by the implicit Crank-Nicholson method for the diffusion terms.
However, the resolution of the equations is not fully made in the wavenumber space, otherwise the calculation of the non-linear convective terms would be very time consuming; in fact the S_i terms contain the $u_i u_j$ products but because of the transformation properties, an algebraic product in physical space corresponds to convolution product in spectral space.
Thus, u_i and u_j are back transformed on physical space, the product is then computed and re-transformed on wavenumber space. During this operation some aliasing problem appears but it is eliminated in a suitable manner.

4.2 Weak numerical scheme for particles equation

Attention is now focused on the time integration of the set of SDEs (operator T). Numerical schemes proposed in this section are directly derived from those proposed by Peirano, Chibbaro, Pozorski, Minier [18]. A proper treatment of the physics of the multiscale aspect imposes to

put forward weak numerical schemes which are consistent with all asymptotic limits when the different time scales go to zero. The development of a suitable weak numerical scheme for the time integration of SDEs is a much more difficult task than the corresponding one for ODEs. Indeed, SDEs do not obey the rules of classical differential calculus, and one has to rely on the theory of stochastic processes. In fact the Wiener process used to model particle velocity is nowhere differentiable, so it's obvious that normal differential calculus rules cannot be adopted. In the present work, Itô's calculus is adopted and therefore all SDEs are written in the *Itô' sense*. This choice has no physical motivation: Itô's calculus is very convenient in the development of weak numerical schemes for SDEs because of the zero mean and isometry properties. We now recall briefly Itô's calculus rules.

In classical integration, the limit of the following sum ($\tau_k \in [t_k, t_{k+1}]$)

$$\int_{t_0}^{t} X(s)\, dW(s) = \lim_{N\to+\infty} \sum_{k=0}^{N} X(\tau_k)(W(t_{k+1}) - W(t_k)), \tag{4.2.1}$$

should be independent of the choice of τ_k. This is not true if we are dealing with Stochastic calculus. As a consequence, a choice has to be made for the sake of consistency. Two main choices (there exist others) are encountered in the literature, the Itô and the Stratonovich definitions. In the Itô definition, $\tau_k = t_k$ and the following limit is under consideration

$$\lim_{N\to+\infty} \sum_{k=1}^{N} X(t_k))(W(t_{k+1}) - W(t_k)). \tag{4.2.2}$$

This choice has a major drawback, i.e. the rules of ordinary differential calculus are no longer valid. However, this drawback is balanced by *the zero mean and isometry properties* which are of great help when deriving weak numerical schemes

$$\begin{aligned} &\langle \int_{t_0}^{t_1} X(s)\, dW(s) \rangle = 0, \\ &\langle \int_{t_0}^{t_2} X(s)\, dW(s) \int_{t_1}^{t_3} Y(s)\, dW(s) \rangle = \int_{t_1}^{t_2} \langle X(s)Y(s) \rangle\, ds. \end{aligned} \tag{4.2.3}$$

where $\langle \cdot \rangle$ is the mathematical expectation ($t_0 \le t_1 \le t_2 \le t_3$, $X(t)$ and $Y(t)$ are two stochastic processes).

In Chapter 2, the correspondence (in a weak sense) between a set of SDEs and a Fokker-Planck equation (for the associated law) has been established. In this work, weak numerical schemes

shall be developed for Eqs. 3.2.1, i.e. we are not interested in the exact trajectories of the process but instead in statistics (the pdf) extracted from the stochastic particles (the real particles are replaced by stochastic ones which should reproduce the same statistics). The numerical method proposed in this work is therefore nothing else than the simulation of an underlying filtered density function, or in other words, the equivalent Fokker-Planck equation is solved by simulating the trajectories of stochastic particles, that is by a dynamical Monte Carlo method.
The weak numerical schemes are developed based on the analytical solution to Eqs. 3.2.1 *with constant coefficients* (independent of time), the main idea being to derive a numerical scheme by freezing the coefficients on the integration intervals. This methodology ensures *stability* and *consistency with all limit systems*:

- stability because the form of the equations gives analytical solutions with exponentials of the type $\exp(-\Delta t/T)$ where T is one of the characteristic time scales (τ_p and $T^*_{L,i}$),
- consistency with all limit systems by construction, since the schemes are based on an analytical solution.

Different techniques shall be used to derive first and second-order (in time) schemes from the analytical solutions with constant coefficients. A first-order scheme can be obtained by computing, at each time step, the variables on the basis of the analytical solutions (all coefficients are frozen at the beginning of the integration interval), i.e. a numerical scheme of the *Euler* kind is obtained. A second-order scheme can be derived by resorting to a predictor-corrector technique where the prediction step is the first-order scheme.
Before presenting the weak numerical schemes, it is a prerequisite to give the analytical solutions to system 3.2.1, with constant coefficients (in time). These solutions are obtained by resorting to Itô's calculus in combination with the method of the variation of the constant. For instance, for the fluid velocity seen, one seeks a solution of the form $U_{s,i}(t) = H_i(t)\exp(-t/T_i)$, where $H_i(t)$ is a stochastic process defined by (indicating $T^*_{L,i}$ with T_i)

$$dH_i(t) = \exp(t/T_i)[C_i\,dt + \check{B}_i\,dW_i(t)], \tag{4.2.4}$$

that is, by integration on a time interval $[t_0, t]$ ($\Delta t = t - t_0$),

$$\begin{aligned} U_{s,i}(t) = U_{s,i}(t_0)\exp(-\Delta t/T_i) + C_i\,T_i\,[1-\exp(-\Delta t/T_i)] \\ +\check{B}_i\exp(-t/T_i)\int_{t_0}^{t}\exp(s/T_i)\,dW_i(s), \end{aligned} \tag{4.2.5}$$

where $\check{B}_i = B_{ii}$ since B_{ij} is a diagonal matrix, cf. Eq. 3.1.3. By proceeding in the same way for the other equations (position and velocity), the analytical solution is obtained for the entire system, cf. Table 4.1.

The three stochastic integrals, Eqs. 4.2.9 to 4.2.11 in Table 4.1, are centred Gaussian processes (they are stochastic integrals of deterministic functions). These integrals are defined implicitly, but they can be simplified by integration by parts, cf. Table 4.1. For the numerical representation of the stochastic integrals, the knowledge of the covariance matrix (second-order moments) is needed, see Table 4.2, as it will be explained in the next section.

Table 4.1: Analytical solutions to system (3.2.1) for time-independent coefficients.

$$x_{p,i}(t) = x_{p,i}(t_0) + U_{p,i}(t_0)\tau_p[1-\exp(-\Delta t/\tau_p)] + U_{s,i}(t_0)\,\theta_i\{T_i[1-\exp(-\Delta t/T_i)] + \tau_p[\exp(-\Delta t/\tau_p)-1]\} + [C_i\,T_i]\{\Delta t - \tau_p[1-\exp(-\Delta t/\tau_p)] - \theta_i(T_i[1-\exp(-\Delta t/T_i)] + \tau_p[\exp(-\Delta t/\tau_p)-1])\} + \Omega_i(t) \quad (4.2.6)$$

$$\text{with} \quad \theta_i = T_i/(T_i-\tau_p)$$

$$U_{p,i}(t) = U_{p,i}(t_0)\exp(-\Delta t/\tau_p) + U_{s,i}(t_0)\,\theta_i[\exp(-\Delta t/T_i) - \exp(-\Delta t/\tau_p)] + [C_i\,T_i]\{[1-\exp(-\Delta t/\tau_p)] - \theta_i[\exp(-\Delta t/T_i) - \exp(-\Delta t/\tau_p)]\} + \Gamma_i(t) \quad (4.2.7)$$

$$U_{s,i}(t) = U_{s,i}(t_0)\exp(-\Delta t/T_i) + C_i\,T_i[1-\exp(-\Delta t/T_i)] + \gamma_i(t) \quad (4.2.8)$$

The stochastic integrals $\gamma_i(t)$, $\Gamma_i(t)$, $\Omega_i(t)$ are given by:

$$\gamma_i(t) = \check{B}_i \exp(-t/T_i)\int_{t_0}^{t}\exp(s/T_i)\,dW_i(s), \quad (4.2.9)$$

$$\Gamma_i(t) = \frac{1}{\tau_p}\exp(-t/\tau_p)\int_{t_0}^{t}\exp(s/\tau_p)\,\gamma_i(s)\,ds, \quad (4.2.10)$$

$$\Omega_i(t) = \int_{t_0}^{t}\Gamma_i(s)\,ds. \quad (4.2.11)$$

By resorting to stochastic integration by parts, $\gamma_i(t)$, $\Gamma_i(t)$, $\Omega_i(t)$ can be written:

$$\gamma_i(t) = \check{B}_i\,\exp(-t/T_i)\,I_{1,i}, \quad (4.2.12)$$

$$\Gamma_i(t) = \theta_i\,\check{B}_i\,[\exp(-t/T_i)\,I_{1,i} - \exp(-t/\tau_p)\,I_{2,i}], \quad (4.2.13)$$

$$\Omega_i(t) = \theta_i\,\check{B}_i\,\{(T_i-\tau_p)\,I_{3,i} - [T_i\exp(-t/T_i)\,I_{1,i} - \tau_p\exp(-t/\tau_p)\,I_{2,i}]\}, \quad (4.2.14)$$

$$\text{with} \quad I_{1,i} = \int_{t_0}^{t}\exp(s/T_i)\,dW_i(s), \quad I_{2,i} = \int_{t_0}^{t}\exp(s/\tau_p)\,dW_i(s), \quad I_{3,i} = \int_{t_0}^{t}dW_i(s).$$

4.3 Limit systems of analytical solution

In limit case 1, where the discrete particles behave as fluid particles, the limit system is given by Eq. 3.2.2. When $\tau_p \to 0$, Eq. 4.2.7 becomes

$$U_{p,i}(t) = U_{s,i}(t_0)\exp(-\Delta t/T_i) + C_i\,T_i\,\exp(-\Delta t/T_i) + \Gamma_i(t), \tag{4.3.1}$$

and for the stochastic integral $\Gamma_i(t)$, one has

$$\begin{cases} \langle \Gamma_i^2(t)\rangle \xrightarrow[\tau_p\to 0]{} \dfrac{\breve{B}_i^2\,T_i}{2}[1-\exp(-2\Delta t/T_i)] = \langle \gamma_i^2(t)\rangle, \\ \langle \Gamma_i(t)\,\gamma_i(t)\rangle \xrightarrow[\tau_p\to 0]{} \langle \gamma_i^2(t)\rangle. \end{cases} \tag{4.3.2}$$

The last two equations indicate that $\Gamma_i(t) \to \gamma_i(t)$ when $\tau_p \to 0$. By comparing Eq. 4.3.1 to Eq. 4.2.8 with $\Gamma_i(t) = \gamma_i(t)$, the results of Eq. 3.2.2 are retrieved, i.e. $\mathbf{U}_p(t) = \mathbf{U}_s(t)$.

In limit case 2, the fluid velocity seen, $\mathbf{U}_s(t)$, is a fast variable which is eliminated. The results obtained in Table 4.1 and 4.2 with $T_i \to 0$ and $\breve{B}_i\,T_i = \text{cst}$, give

$$\begin{aligned} U_{p,i}(t) = U_{p,i}(t_0)\exp(-\Delta t/\tau_p) + [\langle U_i\rangle_L + \mathcal{A}_i\,\tau_p][1-\exp(-\Delta t/\tau_p)] \\ + \sqrt{\frac{\breve{B}_i^2\,T_i^2}{2\tau_p}[1-\exp(-2\Delta t/\tau_p)]}\;\;\mathcal{G}_{p,i}, \end{aligned} \tag{4.3.3}$$

where $\mathcal{G}_{p,i}$ is a $\mathcal{N}(0,1)$ vector (composed of independent standard Gaussian random variables) and we recall that $\langle U_i\rangle_L = \langle U_i(t,\mathbf{x}_p(t)\rangle_L$. It can be rapidly verified, by applying Itô's calculus, that Eq. 4.3.3 is the solution to system 3.2.3 when the coefficients are constant.

In limit case 3, both the fluid velocity seen and the velocity of the discrete particles become rapid variables. When $\tau_p \to 0$ and $T_i \to 0$ with $\breve{B}_i\,T_i = \text{cst}$, Eq. 4.2.6 becomes

$$x_{p,i}(t) = x_{p,i}(t_0) + [\langle U_i\rangle_L + \mathcal{A}_i\,\tau_p]\,\Delta t + \sqrt{\breve{B}_i^2\,T_i^2\,\Delta t}\;\;\mathcal{G}_{x,i}, \tag{4.3.4}$$

which is the solution to Eq. 3.2.4 when the coefficients are constant ($\mathcal{G}_{x,i}$ is a $\mathcal{N}(0,1)$ vector).

In limit case 4, when $T_i \to 0$ (with no condition on $\breve{B}_i T_i$) the system becomes deterministic,

the results are in agreement with Eq. 3.2.5. When $T_i \to 0$, Eqs. 4.2.6 to 4.2.8 become

$$\begin{cases} U_{s,i}(t) = \langle U_i \rangle_L, \\ U_{p,i}(t) = U_{p,i}(t_0)\exp(-\Delta t/\tau_p) + [\langle U_i \rangle_L + \mathcal{A}_i\,\tau_p][1-\exp(-\Delta t/\tau_p)], \\ x_{p,i}(t) = x_{p,i}(t_0) + \tau_p[1-\exp(-\Delta t/\tau_p)]U_{p,i}(t_0) \\ \qquad\qquad + [\langle U_i \rangle_L + \mathcal{A}_i\,\tau_p]\{\Delta t - \tau_p[1-\exp(-\Delta t/\tau_p)]\}, \end{cases} \tag{4.3.5}$$

which is the analytical solution to system 3.2.5 when the coefficients are constant.

4.4 Simulation of a Gaussian vector

Let $\mathbf{X} = (X_1, ..., X_d)$ be a Gaussian vector defined by a zero mean and a covariance matrix $C_{ij} = \langle X_i X_j \rangle$. For all positive symmetric matrix (such as C_{ij}), there exists a triangular matrix P_{ij} which verifies

$$\mathbf{C} = \mathbf{P}\mathbf{P}^t \quad \to C_{ij} = \sum_{k=1}^{d} P_{ik}P_{jk} \tag{4.4.1}$$

$\mathbf{P}$ is given by the Choleski algorithm

$$\begin{aligned} P_{i1} &= \frac{C_{i1}}{\sqrt{C_{11}}}, \quad 1 \le i \le d \\ P_{ii} &= \Big(C_{ii} - \sum_{j=1}^{i-1} P_{ij}\Big)^{1/2}, \quad 1 < i \le d \\ P_{ij} &= \frac{1}{P_{jj}}\Big(C_{ij} - \sum_{k=1}^{j-1} P_{ik}P_{jk}\Big)^{1/2}, \quad 1 < j < i \le d \\ P_{ij} &= 0, \quad i < j \le d \end{aligned}$$

Let $\mathbf{G} = (G_1, ..., G_d)$ be a vector composed of independent $\mathcal{N}(0,1)$ Gaussian random variables, then it can be shown that the vector $\mathbf{Y} = \mathbf{P}\mathbf{G}$ is a Gaussian vector of zero mean and whose covariance matrix is $\mathbf{C} = \mathbf{P}\mathbf{P}^t$. Therefore, $\mathbf{X}$ and $\mathbf{Y}$ are identical, that is

$$\mathbf{X} = \mathbf{P}\mathbf{G} \quad \to X_i = \sum_{k=1}^{d} P_{ik}G_k \tag{4.4.2}$$

Eq. 4.4.2 shows how stochastic integrals, obtained in the analytical solution, can be simulated. From the zero mean property Eq. 4.2.3, it follows that the first order moments are equal to zero.

From the isometry property Eq. 4.2.3 instead we obtain the following equality:

$$\langle\Big(\sum_m g_m(t)\int_{t_0}^{t} f_m(s)dW(s)\Big)^2\rangle = \sum_m g_m^2(t)\int_{t_0}^{t} f_m^2(s)ds + 2\sum_{m<k} g_m(t)g_k(t)\int_{t_0}^{t} f_m(s)f_k(s)ds \quad (4.4.3)$$

where $g_{im}(t)$ and $f_{im}(t)$ are deterministic functions of time. Equation 4.4.3 allows us to derive the covariance matrix, Table 4.2. Once the covariance matrix is known, we can build up the stochastic integrals using the Choleski algorithm, see Table 4.3.

4.5 Weak first order scheme

The derivation of the weak first order scheme is now rather straightforward since the analytical solutions to system 3.2.1 with constant coefficients have been already calculated. Indeed, the Euler scheme (which is a weak scheme of order 1) is simply obtained by freezing the coefficients at the beginning of the time intervals $\Delta t = [t_n, t_{n+1}]$. Let Z_i^n and Z_i^{n+1} be the approximated values of $Z_i(t)$ at time t_n and t_{n+1}, respectively. The Euler scheme is then simply written by using the results of Tables 4.1 and 4.2 as shown in Table 4.3. Before showing that the scheme is consistent with all limit cases, some clarifications must be given. Here, the limit systems are considered in the *discrete sense*. The observation time scale dt has now become the time step Δt. The time scales τ_p and T_i do not go to zero, as in the continuous sense (Chapter 3), but their values, depending on the history of the particles, can be smaller or greater than Δt. The continuous limits, i.e. Eqs. 3.2.2 to 3.2.5, represent a mathematical limit, whereas in the discrete formulation, as we shall see just below, the limit systems correspond to a numerical solution where the ratios $\Delta t/T_i$ and $\Delta t/\tau_p$ become large (the limit systems are obtained by Taylor expansions).

In limit case 1, when $\tau_p \to 0$ in the continuous sense and $\tau_p \ll \Delta t \ll T_i$ in the discrete sense, the numerical scheme gives $U_{p,i}^{n+1} = U_{s,i}^{n+1}$, see Table 4.3, which is consistent with the results of Section 3.2.

In limit case 2, in the continuous sense $T_i \to 0$ and $\check{B}_i T_i = \text{cst}$, that is the fluid velocity seen $\mathbf{U}_s(t)$ becomes a fast variable which is eliminated. In the discrete case, $\mathbf{U}_s(t)$ is simply observed at a time scale which is great compared to its memory, that is $T_i \ll \Delta t \ll \tau_p$, and the numerical

Table 4.2: Derivation of the covariance matrix for constant coefficients.

$$\langle \gamma_i^2(t) \rangle = \check{B}_i^2 \, \frac{T_i}{2} \, [1 - \exp(-2\Delta t / T_i)] \quad \text{where} \quad \check{B}_i^2 = B_{ii}^2 \tag{4.4.4}$$

$$\langle \Gamma_i^2(t) \rangle = \check{B}_i^2 \, \theta_i^2 \left\{ \frac{T_i}{2} [1 - \exp(-2\Delta t / T_i)] - \frac{2\tau_p T_i}{T_i + \tau_p} [1 - \exp(-\Delta t / T_i) \exp(-\Delta t / \tau_p)] \right.$$
$$\left. + \, \frac{\tau_p}{2} [1 - \exp(-2\Delta t / \tau_p)] \right\} \tag{4.4.5}$$

$$\frac{1}{\check{B}_i^2 \, \theta_i^2} \langle \Omega_i^2(t) \rangle = (T_i - \tau_p)^2 \Delta t + \frac{T_i^3}{2} [1 - \exp(-2\Delta t / T_i)] + \frac{\tau_p^3}{2} [1 - \exp(-2\Delta t / \tau_p)]$$
$$- 2T_i^2 (T_i - \tau_p) [1 - \exp(-\Delta t / T_i)] + 2\tau_p^2 (T_i - \tau_p) [1 - \exp(-\Delta t / \tau_p)]$$
$$- 2 \frac{T_i^2 \tau_p^2}{T_i + \tau_p} [1 - \exp(-\Delta t / T_i) \exp(-\Delta t / \tau_p)] \tag{4.4.6}$$

$$\langle \gamma_i(t) \, \Gamma_i(t) \rangle = \check{B}_i^2 \, \theta_i \, T_i \left\{ \frac{1}{2} [1 - \exp(-2\Delta t / T_i)] - \frac{\tau_p}{T_i + \tau_p} [1 - \exp(-\Delta t / T_i) \exp(-\Delta t / \tau_p)] \right\} \tag{4.4.7}$$

$$\langle \gamma_i(t) \, \Omega_i(t) \rangle = \check{B}_i^2 \, \theta_i \, T_i \left\{ (T_i - \tau_p) [1 - \exp(-\Delta t / T_i)] - \frac{T_i}{2} [1 - \exp(-2\Delta t / T_i)] \right.$$
$$\left. + \, \frac{\tau_p^2}{T_i + \tau_p} [1 - \exp(-\Delta t / T_i) \exp(-\Delta t / \tau_p)] \right\} \tag{4.4.8}$$

$$\frac{1}{\check{B}_i^2 \, \theta_i^2} \langle \Gamma_i(t) \, \Omega_i(t) \rangle = (T_i - \tau_p) \{ T_i [1 - \exp(-\Delta t / T_i)] - \tau_p [1 - \exp(-\Delta t / \tau_p)] \}$$
$$- \frac{T_i^2}{2} [1 - \exp(-2\Delta t / T_i)] - \frac{\tau_p^2}{2} [1 - \exp(-2\Delta t / \tau_p)]$$
$$+ T_i \tau_p \, [1 - \exp(-\Delta t / T_i) \exp(-\Delta t / \tau_p)] \tag{4.4.9}$$

scheme yields (see Table 4.3)

$$U_{s,i}^{n+1} = \langle U_i^n \rangle_L + \sqrt{\frac{[\check{B}_i^n]^2 T_i^n}{2}}\, \mathcal{G}_{1,i}, \tag{4.5.1}$$

where $\langle U_i^n \rangle_L = \langle U_i(t_n, \mathbf{x}_p^n) \rangle_L$. The fluid velocity seen becomes a Gaussian random variable, a result which is physically sound since $\mathbf{U}_s(t)$ is observed at time steps which are greater than its memory. This result is in line with that of the model problem presented in Section 3.2. Furthermore, by Taylor expansion, it can be shown that the numerical scheme is consistent with Eq. 4.3.3.

In limit case 3, that is when $1 \ll \Delta t/T_i$ and $1 \ll \Delta t/\tau_p$ (discrete case), one obtains for the velocity of the particles and for the fluid velocity seen (see Table 4.3)

$$\begin{cases} U_{p,i}^{n+1} = \langle U_i^n \rangle_L + \mathcal{A}_i^n\, \tau_p^n + \sqrt{\dfrac{[\check{B}_i^n]^2}{2}} \dfrac{T_i^n}{T_i^n + \tau_p^n} (\sqrt{T_i^n}\, \mathcal{G}_{1,i} + \sqrt{\tau_p^n}\, \mathcal{G}_{2,i}), \\ U_{s,i}^{n+1} = \langle U_i^n \rangle_L + \sqrt{\dfrac{[\check{B}_i^n]^2\, T_i^n}{2}}\ \mathcal{G}_{1,i}. \end{cases} \tag{4.5.2}$$

Once again, $U_{p,i}(t)$, and $U_{s,i}(t)$, which were eliminated in the continuous case, do not disappear. They become Gaussian random variables, a result which is physically sound since these two random variables are observed at time steps which are greater than their respective memories. Moreover, by Taylor expansion, one can show that the numerical scheme is consistent with Eq. 4.3.4.

In limit case 4, $T_i = 0$, and the flow becomes laminar. It can be easily shown that the numerical scheme is consistent with Eqs. 4.3.5. For instance, one has for the fluid velocity $U_{s,i}^{n+1} = \langle U_i^n \rangle_L$, cf. Table 4.3.

The previous results show that the Euler scheme presented in Table 4.3 is consistent with all limit cases. Therefore, the scheme gives numerical solutions which are physically sound, i.e. a consistent representation of the multiscale character of the model is obtained.

4.6 Weak second order scheme

From a formal point of view, weak high-order schemes for a set of SDEs can be derived for any desired accuracy, though this is much more complicated than for ODEs. Such high-order schemes are generally based on truncated stochastic Taylor expansions. These techniques can

Table 4.3: Weak first-order scheme (Euler scheme)

Numerical integration of the system:

$$x_{p,i}^{n+1} = x_{p,i}^n + A_1\, U_{p,i}^n + B_1\, U_{s,i}^n + C_1\, [T_i^n C_i^n] + \Omega_i^n,$$
$$U_{s,i}^{n+1} = U_{s,i}^n\, \exp(-\Delta t/T_i^n) + [T_i^n C_i^n][1 - \exp(-\Delta t/T_i^n)] + \gamma_i^n,$$
$$U_{p,i}^{n+1} = U_{p,i}^n\, \exp(-\Delta t/\tau_p^n) + D_1\, U_{s,i}^n + [T_i^n C_i^n](E_1 - D_1) + \Gamma_i^n.$$

The coefficients $A_1,\ B_1,\ C_1,\ D_1$ and E_1 are given by:

$$A_1 = \tau_p^n\, [1 - \exp(-\Delta t/\tau_p^n)],$$
$$B_1 = \theta_i^n\, [T_i^n (1 - \exp(-\Delta t/T_i^n) - A_1] \quad \text{with} \quad \theta_i^n = T_i^n/(T_i^n - \tau_p^n),$$
$$C_1 = \Delta t - A_1 - B_1,$$
$$D_1 = \theta_i^n [\exp(-\Delta t/T_i^n) - \exp(-\Delta t/\tau_p^n)],$$
$$E_1 = 1 - \exp(-\Delta t/\tau_p^n).$$

The stochastic integrals $\gamma_i^n,\ \Omega_i^n,\ \Gamma_i^n$ are simulated by:

$$\gamma_i^n = P_{11}\, \mathcal{G}_{1,i},$$
$$\Omega_i^n = P_{21}\, \mathcal{G}_{1,i} + P_{22}\, \mathcal{G}_{2,i}$$
$$\Gamma_i^n = P_{31}\, \mathcal{G}_{1,i} + P_{32}\, \mathcal{G}_{2,i} + P_{33}\, \mathcal{G}_{3,i},$$

where $\mathcal{G}_{1,i},\ \mathcal{G}_{2,i},\ \mathcal{G}_{3,i}$ are independent $\mathcal{N}(0,1)$ random variables.

The coefficients $P_{11},\ P_{21},\ P_{22},\ P_{31},\ P_{32},\ P_{33}$ are defined as:

$$P_{11} = \sqrt{\langle (\gamma_i^n)^2 \rangle},$$
$$P_{21} = \frac{\langle \Omega_i^n \gamma_i^n \rangle}{\sqrt{\langle (\gamma_i^n)^2 \rangle}}, \quad P_{22} = \sqrt{\langle (\Omega_i^n)^2 \rangle - \frac{\langle \Omega_i^n \gamma_i^n \rangle^2}{\langle (\gamma_i^n)^2 \rangle}},$$
$$P_{31} = \frac{\langle \Gamma_i^n \gamma_i^n \rangle}{\sqrt{\langle (\gamma_i^n)^2 \rangle}}, \quad P_{32} = \frac{1}{P_{22}} (\langle \Omega_i^n \Gamma_i^n \rangle - P_{21} P_{31}), \quad P_{33} = \sqrt{\langle (\Gamma_i^n)^2 \rangle - P_{31}^2 - P_{32}^2}).$$

not be applied directly in our particular case since neither the unconditional stability nor the consistency in limit cases can be obtained.
Let us consider the following model problem

$$dX_i(t) = \mathfrak{A}_i(\mathbf{X}(t))\,dt + \sum_j \mathfrak{B}_{ij}(\mathbf{X}(t))\,dW_j(t), \tag{4.6.1}$$

where $\mathfrak{B}_{ij}$ verifies the following singular property

$$\sum_k \sum_j \mathfrak{B}_{kj} \frac{\partial \mathfrak{B}_{ij}}{\partial z_k} = 0\ , \quad \forall\ i \tag{4.6.2}$$

It can be shown, for example by stochastic Taylor expansions, that a predictor-corrector scheme of the type

$$\begin{cases} \tilde{X}_i^{n+1} = X_i^n + \mathfrak{A}_i^n\,\Delta t + \sum_j \mathfrak{B}_{ij}^n\,\Delta W_j, \\ X_i^{n+1} = X_i^n + \frac{1}{2}\left(\mathfrak{A}_i^n + \tilde{\mathfrak{A}}_i^{n+1}\right)\Delta t + \sum_j \frac{1}{2}\left(\mathfrak{B}_{ij}^n + \tilde{\mathfrak{B}}_{ij}^{n+1}\right)\Delta W_j, \end{cases} \tag{4.6.3}$$

is a weak second-order scheme ($\tilde{\mathfrak{A}}_i^{n+1} = \mathfrak{A}_i(\tilde{\mathbf{X}}^{n+1})$, $\tilde{\mathfrak{B}}_{ij}^{n+1} = \mathfrak{B}_{ij}(\tilde{\mathbf{X}}^{n+1})$, $\Delta t = t_{n+1} - t_n$ and $\Delta W_j = W_j^{n+1} - W_j^n$). This result is true, once again, only when the diffusion matrix verifies property (4.6.2). Since the predictor step of the scheme above is the Euler scheme (already developed in Section 4.5), the remaining task consists in finding a suitable correction step which ensures the fulfilment of the constraints listed above.
How should the coefficients of the predictor step, $\tilde{\mathfrak{A}}_i^{n+1}$ and $\tilde{\mathfrak{B}}_{ij}^{n+1}$, be computed? The main idea here is to generate a correction step based on the analytical solutions by considering that the acceleration terms vary linearly with time. This idea originates from considerations related to Taylor series expansions. The numerical solution obtained from the analytical solution with constant coefficients is an approximation of first-order accuracy. Mathematically, the solution is given in terms of the integral of acceleration terms. Thus, one can state that the solution based on the zero-th order (constant terms) development of the acceleration terms gives a first-order approximation in time. By analogy, it can be guessed that approximating the acceleration terms by piecewise linear functions in time yields a second-order approximation in time.

Let us introduce the following notation: $\tilde{U}_{p,i}^{n+1}$ and $\tilde{U}_{s,i}^{n+1}$ stand for the predicted velocities and $\tilde{T}_i^{n+1}$ and $\tilde{\tau_p}^{n+1}$ are the predicted time scales. The values of the fields related to the fluid

taken at $(t_{n+1},\, \mathbf{x}_p^{n+1})$ are denoted, for example, $\langle U_i^{n+1}\rangle_L$ or $\langle P^{n+1}\rangle_L$. As far as the computation of the mean fields extracted from the discrete particles are concerned, it is worth emphasising that none of them are computed at $(t_{n+1},\, \mathbf{x}_p^{n+1})$, because the scheme would become implicit, i.e. fields such as the expected value of the particle velocity are computed from the predicted velocities. For example, one has

$$C_i(t_{n+1},\, \mathbf{x}_p^{n+1}) = C_i^{n+1} = \frac{\langle U_i^{n+1}\rangle_L}{\tilde{T}_i^{n+1}} + f(\langle \tilde{\mathbf{U}}_p^{n+1}\rangle_L, \langle \mathbf{U}^{n+1}\rangle_L, \langle P^{n+1}\rangle_L). \tag{4.6.4}$$

Let us first consider the fluid velocity seen. The analytical solution to system (3.2.1) when the coefficients are constant is, by applying the rules of Itô's calculus

$$U_{s,i}(t) = U_{s,i}(t_0)\, \exp(-\Delta t/T_i) + \int_{t_0}^{t} C_i(s, \mathbf{x}_p) \exp[(s-t)/T_i]\, ds + \gamma_i(t), \tag{4.6.5}$$

where the temporal coefficients (the time scales) are considered constant, while the term $C_i(s, \mathbf{x}_p)$ is retained in the integral. Following the previous ideas, let us suppose that $C_i(s, \mathbf{x}_p)$ varies linearly on the integration interval $[t_0, t]$, that is $(\Delta t = t - t_0)$

$$C_i(s, \mathbf{x}_p(s)) = C_i(t_0, \mathbf{x}_p(t_0)) + \frac{1}{\Delta t}[C_i(t, \mathbf{x}_p(t)) - C_i(t_0, \mathbf{x}_p(t_0))](s - t_0). \tag{4.6.6}$$

By inserting Eq. (4.6.6) into Eq. (4.6.5), integration gives

$$\begin{aligned} U_{s,i}(t) = {} & U_{s,i}(t_0)\, \exp(-\Delta t/T_i) + [T_i\, C_i(t_0, \mathbf{x}_p(t_0))]\, A_2(\Delta t, T_i) \\ & + [T_i\, C_i(t, \mathbf{x}_p(t))]\, B_2(\Delta t, T_i) + \gamma_i(t), \end{aligned} \tag{4.6.7}$$

where the functions $A_2(\Delta t, x)$ and $B_2(\Delta t, x)$ are given by (x is a positive real variable)

$$\begin{cases} A_2(\Delta t, x) = -\exp(-\Delta t/x) + [1 - \exp(-\Delta t/x)][\Delta t/x], \\ B_2(\Delta t, x) = 1 - [1 - \exp(-\Delta t/x)][\Delta t/x]. \end{cases} \tag{4.6.8}$$

Accounting for the time dependence of the coefficients, i.e. T_i, it is proposed to write the following correction step, which is in line with the treatment of the acceleration terms,

$$\begin{aligned} U_{s,i}^{n+1} = {} & \frac{1}{2}\, U_{s,i}^{n}\, \exp(-\Delta t/T_i^n) + \frac{1}{2}\, U_{s,i}^{n}\, \exp(-\Delta t/\tilde{T}_i^{n+1}) \\ & + A_2(\Delta t,\, T_i^n)\, [T_i^n C_i^n] + B_2(\Delta t,\, \tilde{T}_i^{n+1})\, [\tilde{T}_i^{n+1} C_i^{n+1}] + \tilde{\gamma}_i^{n+1}, \end{aligned} \tag{4.6.9}$$

where a consistent formulation for the stochastic integral $\tilde{\gamma}_i^{n+1}$ is needed. The same procedure is used, i.e. the diffusion matrix B_{ij} is linearised and integration is carried out. The final expression is

$$\tilde{\gamma}_i^{n+1} = \sqrt{[B_i^*]^2 \frac{\tilde{T}_i^{n+1}}{2}[1 - \exp(-2\Delta t/\tilde{T}_i^{n+1})]}\ \ \mathcal{G}_{1,i}, \tag{4.6.10}$$

where $\mathcal{G}_{1,i}$ is the same $\mathcal{N}(0,1)$ random variable used in the simulation of γ_i^n in the Euler scheme and where B_i^* is defined by

$$\begin{aligned}\Big[1 - \exp(-2\,\Delta t/\tilde{T}_i^{n+1})\Big] B_i^* = \\ A_2(2\,\Delta t,\, \tilde{T}_i^{n+1})\,\sqrt{(\check{B}_i^n)^2} + B_2(2\,\Delta t,\, \tilde{T}_i^{n+1})\,\sqrt{(\tilde{\check{B}}_i^{n+1})^2}.\end{aligned} \tag{4.6.11}$$

In the case of the velocity of the particles, the same approach followed for the fluid velocity seen is adopted. It is worth emphasising that no correction is done on position, $\mathbf{x}_p(t)$, since the prediction is already of order 2. The complete scheme is summarised in Table 4.4.

Table 4.4: Weak second-order scheme

Prediction step: Euler scheme, see Table 4.3.

(4.6.12)

Correction step:

$$
\begin{aligned}
U_{p,i}^{n+1} &= \frac{1}{2}\,U_{p,i}^{n}\,\exp(-\Delta t/\tau_p^n) + \frac{1}{2}\,U_{p,i}^{n}\,\exp(-\Delta t/\tilde{\tau}_p^{n+1}) \\
&\quad + \frac{1}{2}\,U_{s,i}^{n}\,C_{2c}(\tau_p^n,\,T_i^n) + \frac{1}{2}\,U_{s,i}^{n}\,C_{2c}(\tilde{\tau}_p^{n+1},\,\tilde{T}_i^{\,n+1}) \\
&\quad + A_{2c}(\tau_p^n,\,T_i^n)\,[T_i^n C_i^n] + B_{2c}(\tilde{\tau}_p^{n+1},\,\tilde{T}_i^{n+1})\,[\tilde{T}_i^{n+1} C_i^{n+1}] \\
&\quad + A_2(\Delta t, \tau_p^n)[\tau_p^n\,\mathcal{A}_i^n] + B_2(\Delta t, \tilde{\tau}_p^{\,n+1})[\tilde{\tau}_p^{\,n+1}\,\mathcal{A}_i^{n+1}] + \tilde{\Gamma}_i^{n+1}, \\
U_{s,i}^{n+1} &= \frac{1}{2}\,U_{s,i}^{n}\,\exp(-\Delta t/T_i^n) + \frac{1}{2}\,U_{s,i}^{n}\,\exp(-\Delta t/\tilde{T}_i^{n+1}) + A_2(\Delta t,\,T_i^n)\,[T_i^n C_i^n] \\
&\quad + B_2(\Delta t,\,\tilde{T}_i^{n+1})\,[\tilde{T}_i^{n+1} C_i^{n+1}] + \tilde{\gamma}_i^{n+1}.
\end{aligned}
$$

(4.6.13)

The coefficients $A_2,\ B_2,\ A_{2c},\ B_{2c}$ et C_{2c} are defined as:

$$
\begin{aligned}
&A_2(\Delta t, x) = -\exp(-\Delta t/x) + [1-\exp(-\Delta t/x)][\Delta t/x], \\
&B_2(\Delta t, x) = 1 - [1-\exp(-\Delta t/x)][\Delta t/x], \\
&A_{2c}(x,y) = -\exp(-\Delta t/x) + [(x+y)/\Delta t][1-\exp(-\Delta t/x)] - (1+y/\Delta t)\,C_{2c}(x,y), \\
&B_{2c}(x,y) = 1 - [(x+y)/\Delta t][1-\exp(-\Delta t/x)] + (y/\Delta t)\,C_{2c}(x,y), \\
&C_{2c}(x,y) = [y/(y-x)][\exp(-\Delta t/y) - \exp(-\Delta t/x)].
\end{aligned}
$$

(4.6.14)

The stochastic integrals $\tilde{\gamma}_i^{n+1}$ and $\tilde{\Gamma}_i^{n+1}$ are simulated as follows:

$$
\begin{aligned}
&\tilde{\gamma}_i^{n+1} = \sqrt{\frac{[B_i^*]^2\tilde{T}_i^{n+1}}{2}[1-\exp(-2\Delta t/\tilde{T}_i^{n+1})]}\;\mathcal{G}_{1,i}, \\
&\text{with}\quad \left[1-\exp(-2\,\Delta t/\tilde{T}_i^{n+1})\right]\; B_i^* = A_2(2\,\Delta t,\,\tilde{T}_i^{n+1})\,\sqrt{(\check{B}_i^n)^2} + \\
&\qquad\qquad B_2(2\,\Delta t,\,\tilde{T}_i^{n+1})\,\sqrt{(\tilde{\check{B}}_i^{n+1})^2}. \\
&\tilde{\Gamma}_i^{n+1} = \frac{\langle\tilde{\Gamma}_i^{n+1}\tilde{\gamma}_i^{n+1}\rangle}{\langle(\tilde{\gamma}_i^{n+1})^2\rangle}\,\tilde{\gamma}_i^{n+1} + \sqrt{\langle(\tilde{\Gamma}_i^{n+1})^2\rangle - \frac{\langle\tilde{\Gamma}_i^{n+1}\tilde{\gamma}_i^{n+1}\rangle^2}{\langle(\tilde{\gamma}_i^{n+1})^2\rangle}}\;\mathcal{G}_{2,i} \\
&\text{with}\quad \langle\tilde{\Gamma}_i^{n+1}\tilde{\gamma}_i^{n+1}\rangle = \langle\Gamma_i\gamma_i\rangle(\tau_p^n,\,\tilde{T}_i^{n+1},\,B_i^*)\quad\text{and}\quad \langle(\tilde{\Gamma}_i^{n+1})^2\rangle = \langle\Gamma_i^2\rangle(\tau_p^n,\,\tilde{T}_i^{n+1},\,B_i^*).
\end{aligned}
$$

Chapter 5

Implementation and validation

Since we are concerned with a hybrid model which uses the output coming from the LES solver (FLOWSB) for particle tracking, a brief description of the structure of the FLOWSB code is useful to understand the interconnection between FLOWSB and the implemented stochastic module. A picture of the structure of FLOWSB is given, included the "old" particle tracker with no model. Then the main new routines and their position in the code are briefly described , with particular attention to the exchanged information between the LES solver and the stochastic module.

5.1 Structure of FLOWSB

The subroutine *Buoyan* open the files needed to write or from which it reads the informations necessary for the restart of the simulation.
The subroutine *Supvis* reads from the *ft5* file the input parameters for the simulation: number of grid nodes, Reynolds number, subgrid model for LES and other technical parameters. If the flag PART is equal to one, the Lagrangian tracker will be executed so that the next step will be the call to the subroutine Partinput.
The subroutine *Partinput* reads from the input file *inputpart* the parameters necessary for particle tracking: number of particles in the domain, number of set of particles, frequency of tracking, order of interpolation, etc.
The subroutine *Partinit* initialize particles positions and velocities in the field: particles positions are initialized randomly, while particles velocities are imposed equal to the velocity of the fluid

at particle position. In the case of a restart for particles, their positions and velocities are read from a restart file *Readrestart*.

Then is called *Excute*, which is the main subroutine of FLOWSB and which provides the time advancement of the flow field through a pseudo-spectral method (the calls to all the subroutines in Excute are omitted because not relevant for our purpose). At the end of the time advancement of the flow field, in Excute is called the subroutine *Parttrack* which provides the Lagrangian particle tracking. The "old" particle tracker with no model used the velocity field interpolated at particle position to evaluate the drag force on each particle (also the other forces if requested, i.e. basset force, lift, etc.) and then used a fourth order Runge-Kutta scheme for the time advancement. Terminated *Parttrack*, we return back to *Excute* which begins the next step and so on. So the subroutines that has been changed are all the ones mentioned above concerned with particles, while all the others have remained the same.

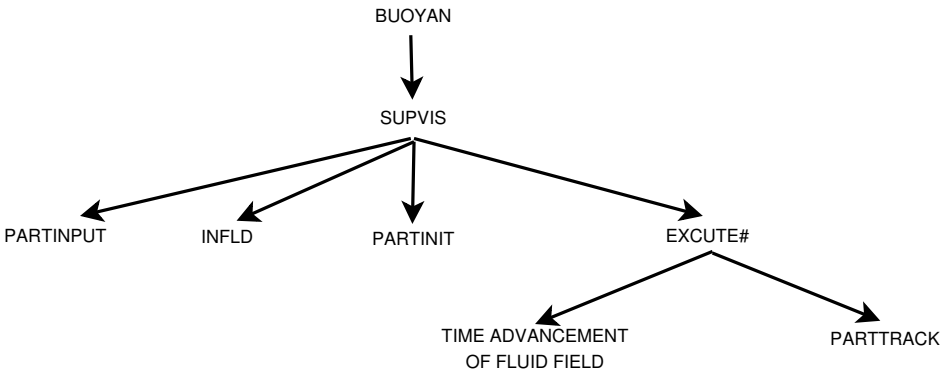


Figure 5.1: Structure of the FLOWSB code

5.2 Stochastic module

We show now the main changes applied to the code following the same order in which the subroutines are called in FLOWSB.

Partinit

In the subroutine *Partinit* the initialization of the variable U_s has been added, which was not present previously because it is a new variable of this model. In the case of a new simulation, for each particle, U_s is imposed equal to the velocity of the fluid field interpolated at particle

position, while in the case of a restart it is read from a particle restart file. In this subroutine the Gaussian vector and in particular its seed for the generation of Gaussian random numbers is also initialized. The random numbers generator has been taken from a library but we need to pay some attention to its correct use. A first and unique call to the subroutine *Zufalli* provides the initialization of the seed, and then at each time step a call to the subroutine *Normalen* generates a vector, of the same length of the number of particles, of random numbers picked from a Gaussian distribution of zero mean and unit variance. In the case of a restart, the call to *Zufalli* is no more needed, but it's necessary to read some parameters from the restart file before calling the subroutine *Normalen*. These new quantities cited above are passed to the particle tracker through a common block.

Parttrack

The subroutine *Parttrack* has been completely replaced because nothing was in common with the old one. In Figure 5.2 the structure of the subroutine *Parttrack* is sketched with all its own

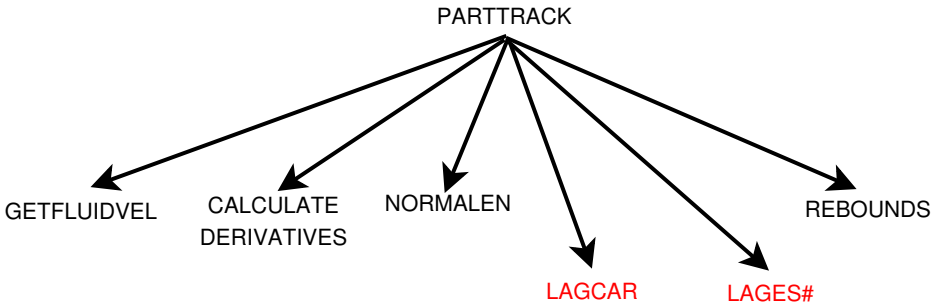


Figure 5.2: Structure of the subroutine parttrack

subroutines called in the order from left to right. First of all the subroutines *Getfluidvel* and *Calculatederivatives* are called, which convert the fluid velocity and velocity derivatives from spectral to physical space and return as outputs the velocity field at grid nodes and the derivatives of all velocity components in all three directions at grid nodes. This fluid field is the one just advanced in time, so at time t_{n+1}. For the first order scheme discussed in the previous chapter, we need fluid quantities frozen at time t_n, so it is necessary at the end of each time step to store all fluid quantities in vectors like $UOLD$ (the same for derivatives), which will be used in the next step for tracking. Fluid quantities at time t_{n+1} are used only in the second order scheme for the correction step.

Next, we find the call to *Normalen*, which as previously said, generates a vector of random numbers picked from a Gaussian distribution.
The subroutine *Lagcar* provides the calculation of the diffusion coefficient, say $BX(nset, n, i)$ and of all the large scale terms appearing in the stochastic equation for U_s, stored in the vector $PIIL(nset, n, i)$. In particular in $PIIL$ we store the term $-\frac{1}{\rho_f}\frac{\partial\langle p\rangle_L}{\partial x_i}\,dt+(\langle U_{p,j}\rangle_L - \langle U_j\rangle_L)\,\frac{\partial\langle U_i\rangle_L}{\partial x_j}\,dt$ in Eq. 3.1.3, which in the discretized equations corresponds to the term $T_i^n C_i^n$ in Table 4.3. The following procedure is adopted to evaluate these terms, see Berrouk *et al.*[19]

- Known the velocity and the velocity derivatives from LES solver, subgrid dissipation and kinetic energy are evaluated through the relations

$$\epsilon_r = -\tau_{ij}\frac{\partial\langle U_i\rangle_L}{\partial x_j} = (C_S\Delta)^2|\mathcal{S}|^3 \tag{5.2.1}$$

and

$$k_{SGS} = C_\epsilon(\Delta\epsilon_r)^{2/3} \tag{5.2.2}$$

where $C_\epsilon \approx 1$ is a model constant and $C_S \approx 0.065$ is the Smagorinsky constant (see Kim & Moin [17]) and Δ is the grid spacing. The second equality in Eq. 5.2.1 has been obtained using the Smagorinsky model for τ. $\mathcal{S}$ is the module of the resolved rate of strain tensor

$$|\mathcal{S}| = |2\langle S_{ij}\rangle_L\langle S_{ij}\rangle_L|^{1/2} \qquad \text{with} \qquad \langle S_{ij}\rangle_L = \frac{1}{2}\Big(\frac{\partial\langle U_i\rangle_L}{\partial x_j} + \frac{\partial\langle U_j\rangle_L}{\partial x_i}\Big) \tag{5.2.3}$$

- The quantities $\langle U_p\rangle_L$ and $\langle U_r\rangle_L$ are evaluated by the subroutine *Lagstat* accordingly to Eq. 4.0.1, i.e. for each cell of the domain the above quantities are summed for each particle within the cell and then we divide for the number of particles in the cell. In this way we recover LES Eulerian quantities.

- The Lagrangian time scale T_L is computed by the relation

$$T_L = \Big(\frac{1}{2} + \frac{3}{4}C_0\Big)^{-1}\frac{k_{SGS}}{\epsilon_r} \tag{5.2.4}$$

Then the modified time scale T_i is computed by multiplying T_L by the Csanady's factor

$$T_i = T_L\left(1 + \beta_i^2\frac{|\langle\mathbf{U}_r\rangle_L|^2}{2k_{SGS}/3}\right)^{-1/2}, \tag{5.2.5}$$

- The diffusion coefficient attached to each particle is computed through the relation

$$BX_i = \sqrt{\epsilon_r \left(C_0 b_i + \frac{2}{3}(b_i - 1) \right)} \tag{5.2.6}$$

  where b_i is simply $b_i = T_L/T_i$.

- Finally, the term $PIIL$, which contains the gradient of pressure and the term coming from the crossing trajectory effect, is computed for each particle.

All the quantities recalled above are intended to be interpolated at particle position in case of interpolation. For further explaining about the different types of interpolation techniques adopted see Chapter 6.

The next subroutine called in *Parttrack* is *Lages*, which provides time advancement of particle velocity and position through the numerical scheme showed in the previous chapter, using quantities calculated in *Lagcar*. The term $T_i C_i$ appearing in the numerical scheme in Table 4.3 is assembled multiplying the previous calculated $PIIL$ by T_i and summing the fluid velocity at particle position. Then the Stokes number of particles is calculated including the Reynolds number dependence

$$\tau_p^+ = \frac{St}{(1 + 0.15Re^{0.687})} \tag{5.2.7}$$

Finally we assemble all terms necessary for the time advancement scheme, see Table 4.3, and we obtain particles position, velocity and velocity seen at t_{n+1}.

Going back to *Parttrack*, the last call at each time step is to the subroutine *Rebounds* which calculates rebounds for particles that at time t_{n+1} would be out of the domain. For particles leaving the domain in the homogeneous directions, i.e. x and y, we simply relocate them at the beginning of the domain ensuring periodic boundary conditions. Instead, for particles that would be beyond channel walls, i.e. z direction, we apply a condition of elastic rebound for particle position and velocity, while we assign to U_s the velocity of the fluid from the LES solver interpolated at new particle position.

If we are dealing with the second order scheme for time advancement, the procedure is a bit more articulated: up to *Lagcar* it is exactly the same, but then instead of calling *Lages* we call *Lages2* twice. The first time is for the prediction step and to store some terms necessary for the correction step (the terms that in Table 4.4 appear with the apex ^{n}). The second call is for the correction step, using fluid field from LES at t_{n+1}. The position is not corrected because it is already computed with a second order accuracy.

For a better comprehension of the implementation work see Appendix B where the main subroutines listed above are shown.

5.3 Mathematical validation

After the classical checks by printing quantities to see if every passage in the calculus was done correctly, a stronger mathematical validation has been done. We ask, in fact, the numerical solution to be consistent with the analytical solution in case of constant coefficients. We use also this test case with constant coefficients to check if the numerical schemes are consistent with the limit cases.

5.3.1 Preliminary study

Let us consider an isotropic case where $C_i(t, \mathbf{x}_p) = 0$ and where the initial conditions are given by $x_p(0) = U_p(0) = U_s(0) = 0$. In this case the system,

$$\begin{cases} dx_p(t) = U_p\, dt \\ dU_p(t) = \dfrac{1}{\tau_p}(U_s - U_p)\, dt \\ dU_s(t) = -\dfrac{1}{T}U_s\, dt + \sigma dW(t) \end{cases} \tag{5.3.1}$$

admits $x_p(t) = \Omega(t)$, $U_p(t) = \Gamma(t)$ and $U_s(t) = \gamma(t)$ as a solution, see Table 4.1. These random variables are Gaussian and they have zero mean and known variances, see Table 4.2. As a matter of fact, all second order moments (variances) are increasing functions of time with the following asymptotes

$$\begin{cases} \langle \Omega^2(t) \rangle \xrightarrow[t\to\infty]{} \sigma^2\theta^2(T-\tau_p)^2 t + B \\ \langle \Gamma^2(t) \rangle \xrightarrow[t\to\infty]{} \dfrac{\sigma^2 T}{2}\dfrac{T}{T+\tau_p} \\ \langle \gamma^2(t) \rangle \xrightarrow[t\to\infty]{} \dfrac{\sigma^2 T}{2} \end{cases} \tag{5.3.2}$$

where the constant B is defined as

$$B = \frac{\sigma^2\theta^2}{2(T+\tau_p)}\Big(T^3 + \tau_p^3 - 4(T-\tau_p)^2(T+\tau_p) - 4\frac{T^2\tau_p^2}{T+\tau_p}\Big) \tag{5.3.3}$$

In the stationary case, that is as time goes to infinity, the velocities become stationary processes, and one can verify the well-known Tchen formulas which give a relation between the energy of

the particles, the energy of the fluid seen and the velocity covariance

$$\frac{\langle U_p^2(t)\rangle}{\langle U_s^2(t)\rangle} = \frac{T}{T+\tau_p} \qquad \text{and} \qquad \langle U_p(t)U_s(t)\rangle = \langle U_p^2(t)\rangle \tag{5.3.4}$$

The equality, in the stationary case, between the energy of the particles and the velocity covariance, can be easily obtained noticing that

$$\langle \Gamma(t)\gamma(t)\rangle \xrightarrow[t\to\infty]{} \frac{\sigma^2 T}{2}\frac{T}{T+\tau_p} \tag{5.3.5}$$

5.3.2 Limit cases

We now verify that the two numerical schemes (first and second order) fulfill the above requirements, that is (i) the numerical scheme gives the exact solution when the coefficients are constant and (ii) the numerical scheme is consistent in the limit cases. The values taken by the (constant) coefficients in the different cases are presented in Table 5.1. It can be seen that the time step is constant and therefore the separation of scales is obtained by varying the coefficients: in this numerical example, it is considered that there is a separation of scales when the ratio between the smallest one and the greatest one is roughly 10^{-2}.

In practice, the code is modified in order to simulate the trajectories in a domain which can

case	$\tau_p(s)$	$T(s)$	$\sigma(m/s^{3/2})$	$\Delta t(s)$
general case	10^{-1}	$2\cdot 10^{-1}$	10^{1}	10^{-3}
limit case (i)	10^{-5}	10^{-1}	10^{1}	10^{-3}
limit case (ii)	10^{-1}	10^{-5}	10^{3}	10^{-3}
limit case (iii)	$2\cdot 10^{-5}$	10^{-5}	10^{3}	10^{-3}
limit case (iv)	10^{-1}	10^{-15}	10^{1}	10^{-3}

Table 5.1: Numerical simulation for system 5.3.1 in the general and limit cases. General case: $\Delta t \ll T$, τ_p. Limit case (i): $\tau_p \ll \Delta t \ll T$. Limit case (ii): $T \ll \Delta t \ll \tau_p$. Limit case (iii): T, $\tau_p \ll \Delta t$. Limit case (iv): $T \to 0$.

be considered as infinite. Since the purpose of the simulations is not the study of the statistical error, only 6000 trajectories are simulated. It is observed that there is no difference between the numerical solutions given by the first order and the second order scheme. This is not surprising since, with constant coefficients, the correction step in the second order scheme is rigorously the Euler scheme. The results are now displayed for the second order moments, the first order moments are generally omitted since the solutions are Gaussian random variables of zero mean. In the general case, Fig. 5.3, when $\Delta t \ll \tau_p$ and $\Delta t \ll T$, it can be seen that both schemes are

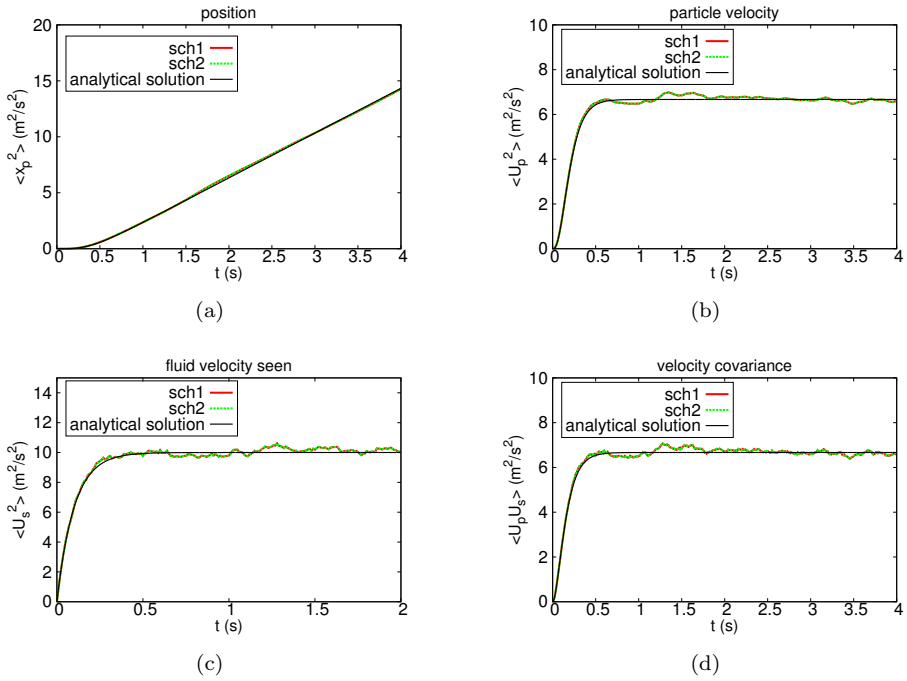


Figure 5.3: General case with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 2 \cdot 10^{-1}s$, $\tau_p = 10^{-1}s$, $\sigma = 10^{1}m/s^{3/2}$

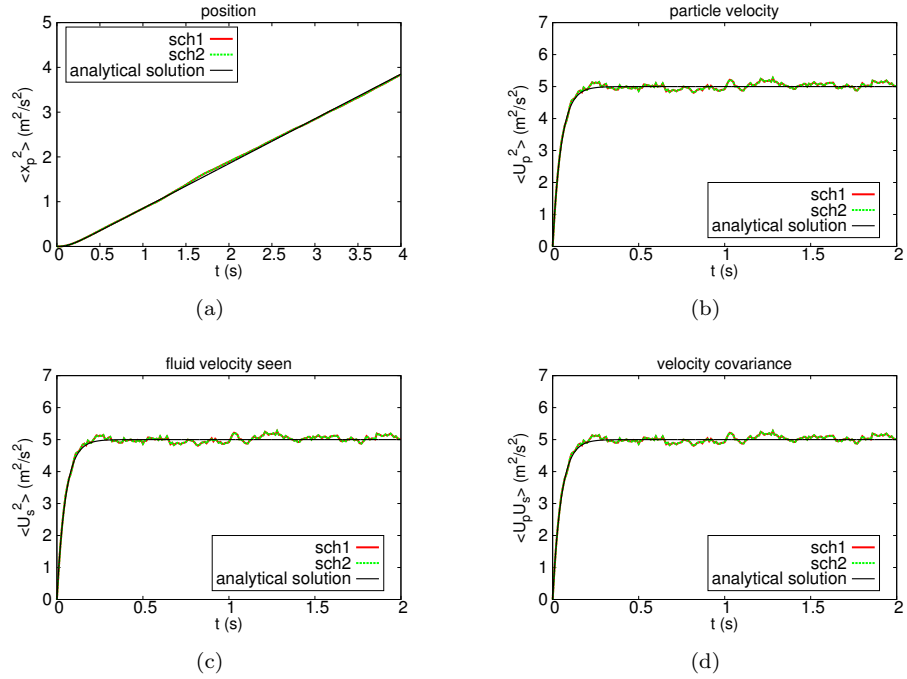


Figure 5.4: Limit case (i) with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 10^{-1}s$, $\tau_p = 10^{-5}s$, $\sigma = 10^{1}m/s^{3/2}$

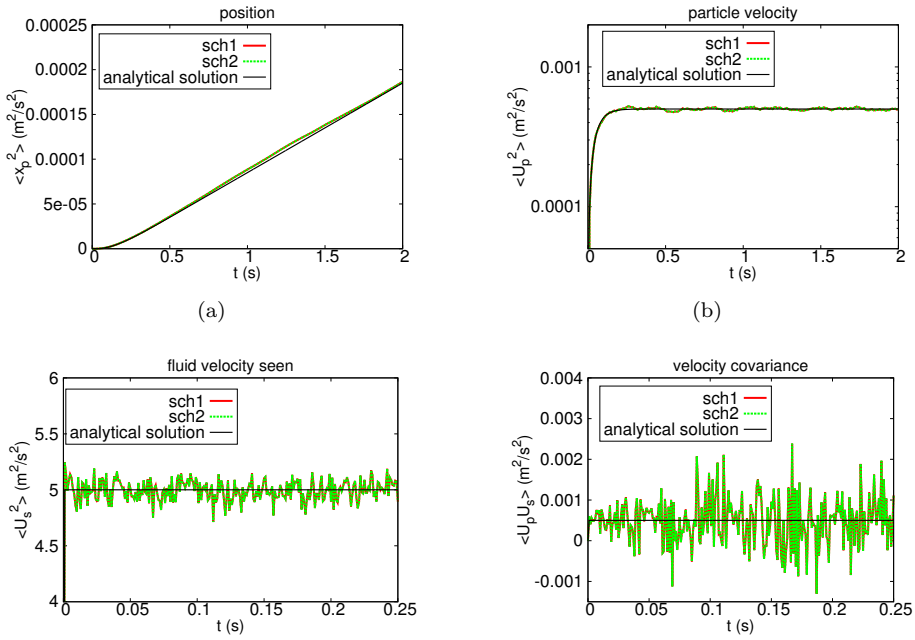


Figure 5.5: Limit case (ii) with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 10^{-5}s$, $\tau_p = 10^{-1}s$, $\sigma = 10^3 m/s^{3/2}$

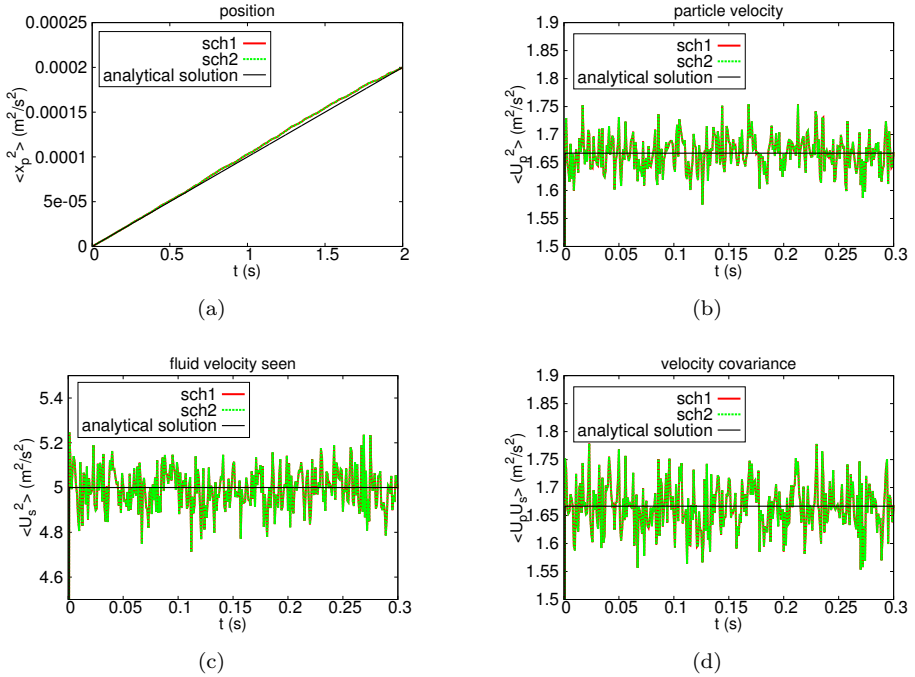


Figure 5.6: Limit case (iii) with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 10^{-5}s$, $\tau_p = 2 \cdot 10^{-5}s$, $\sigma = 10^3 m/s^{3/2}$

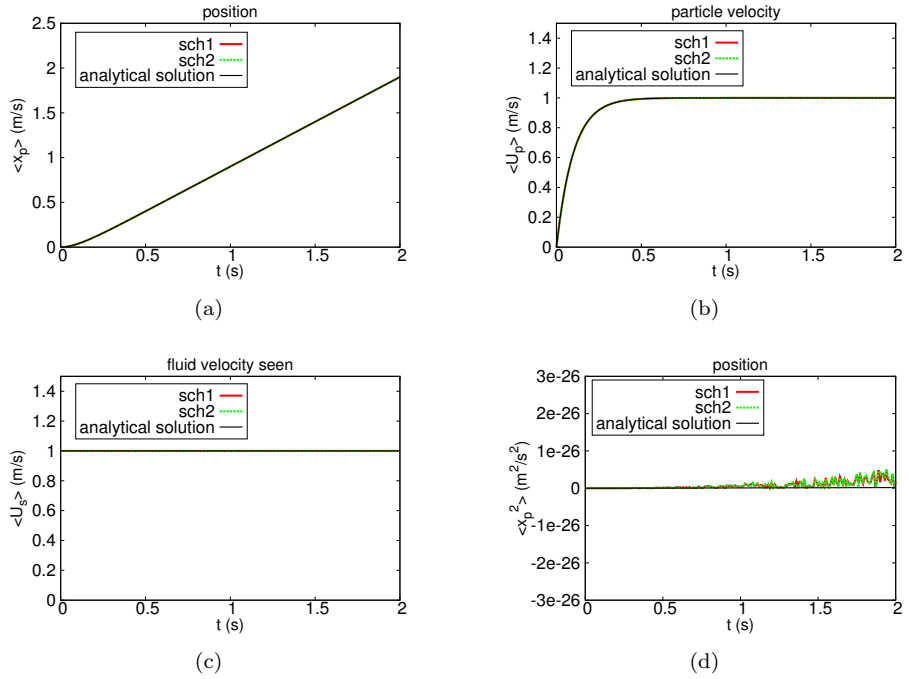


Figure 5.7: Limit case (iv) with $N = 6000$: $\Delta t = 10^{-3}s$, $T = 10^{-15}s$, $\tau_p = 10^{-1}s$, $\sigma = 10^{1}m/s^{3/2}$

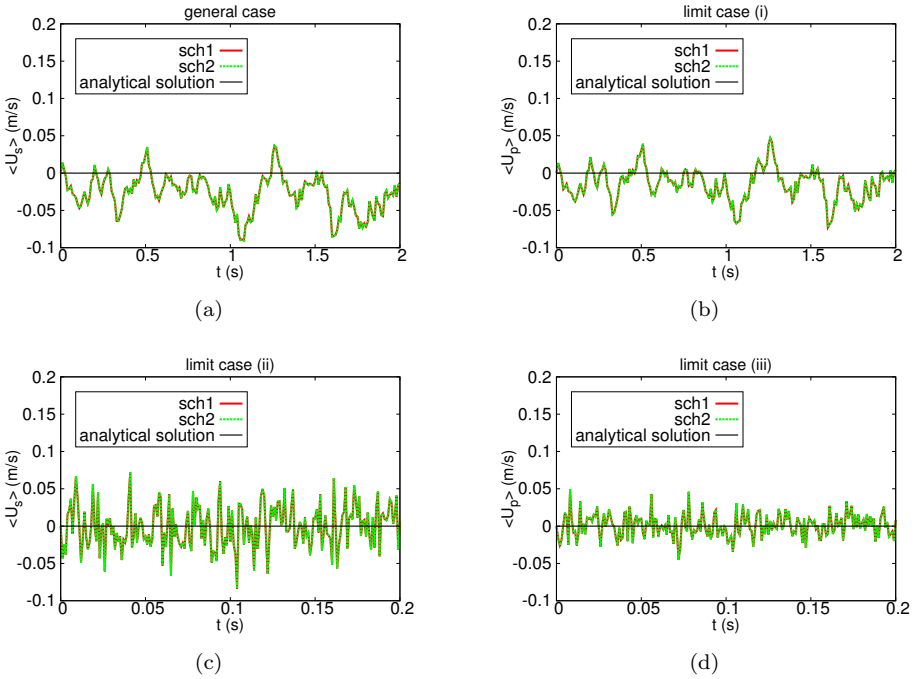


Figure 5.8: Behaviour of first order moments in different cases.

in agreement with the analytical solutions. One can verify that the limit values for $\langle U_p^2(t)\rangle$ and $\langle U_s^2(t)\rangle$ are correct, that is $\sigma^2 T/2 = 10$ and $(\sigma^2 T/2)[T/(T+\tau_p)] = 6.67$, respectively, see Eq. 5.3.2. The results are also in line with the Tchen formulas, $\langle U_p^2(t)\rangle/\langle U_s^2(t)\rangle = T/(T+\tau_p) = 2/3$ and $\langle U_p(t)U_s(t)\rangle = \langle U_p^2(t)\rangle$. Furthermore, the limit behaviour of $\langle x_p^2(t)\rangle$ is correct since the slope of the asymptote is $(\sigma T)^2 = 4$.

For limit case (i), Fig. 5.4, the fluid velocity seen and the particle velocity become identical. The limit values for $\langle U_p^2(t)\rangle$ and $\langle U_s^2(t)\rangle$ are $\sigma^2 T/2 = 5$, and for $\langle x_p^2(t)\rangle$ the slope of the asymptote is $(\sigma T)^2 = 1$. The Tchen formulas are verified, $\langle U_p^2(t)\rangle/\langle U_s^2(t)\rangle = T/(T+\tau_p) \simeq 1$ and $\langle U_p(t)U_s(t)\rangle = \langle U_p^2(t)\rangle$.

For limit case (ii), Fig. 5.5, the fluid velocity seen becomes a fast variable, that is, in the discrete case, $U_s(t)$ is a Gaussian random variable of zero mean and variance $\sigma^2 T/2 = 5$. The asymptote for $\langle U_p^2(t)\rangle$ is $(\sigma T)^2/(2\tau_p) = 5 \cdot 10^{-4}$, which is in agreement with Eq. 5.3.2. The slope of the asymptote for $\langle x_p^2(t)\rangle$ is $(\sigma T)^2 = 10^{-4}$. As far as the Tchen formulas are concerned, it is verified that $\langle U_p^2(t)\rangle/\langle U_s^2(t)\rangle = T/(T+\tau_p) \simeq 10^{-4}$ and $\langle U_p(t)U_s(t)\rangle = \langle U_p^2(t)\rangle$.

In the diffusive regime, limit case (iii), Fig. 5.6, both the fluid velocity seen, $U_s(t)$, and the particle velocity, $U_p(t)$, become fast variables, which in the discrete case gives two Gaussian random variables of zero mean and of variances $\sigma^2 T/2 = 5$ and $(\sigma T)^2[T/(T+\tau_p)] = 1.67$, respectively. For $\langle x_p^2(t)\rangle$, the slope of the asymptote is indeed given by $(\sigma T)^2 = 10^{-4}$. The Tchen formulas are also verified.

At last, in the laminar case, limit case (iv), Fig. 5.7 shows a laminar flow with an imposed mean fluid velocity, $\langle U_s(t)\rangle = 1m/s$. It is seen that the asymptotes obtained for $\langle U_p(t)\rangle$ and $\langle x_p(t)\rangle$ are in agreement with the results derived in Eq. 4.3.5, $\langle U_p(t)\rangle \to \langle U_s(t)\rangle$ and $\langle x_p(t)\rangle \to \langle x_p(t)\rangle = t - 0.1$ when $t \to \infty$. Furthermore, the numerical behaviour of the variances shows that the dynamics of the particles is deterministic.

Finally, as far as the first order moments are concerned, Fig. 5.8 shows that the numerical values obtained are in line with the zero mean result.

5.4 Discretization error

We limit ourselves here to the study of the time discretization error, to show that the first- and second-order numerical scheme developed above, have the expected order of convergence. It's necessary, first of all, to recall the different sources of error occurring in a stochastic particle-mesh method.

The total error depends on three numerical parameters: Δt for time discretization, Δx for space discretization and N for the evaluations of statistical quantities by a finite set of particles. As explained by Chibbaro *et al.*[20], the total error (for $\langle f(\mathbf{Z})\rangle$, the expected value of a functional of $\mathbf{Z}$) is, at time $t = T$

$$\epsilon(T) = \langle f(\mathbf{Z}_T)\rangle - \{f(\overline{\mathbf{Z}}_T)\}_{N,\Delta x} \tag{5.4.1}$$

where $\overline{\mathbf{Z}}_T$ is the approximation of $\mathbf{Z}_T$ after time integration (operator T in Chapter 4) and $\{\,\}_{N,\Delta x}$ is the approximation of the expected value. The total error, which is a random variable, can thus be decomposed as follows

$$\epsilon(T) = \epsilon_N(T) + \epsilon_{\Delta t}(T) + \epsilon_{\Delta x}(T) + \epsilon_\infty(T) \tag{5.4.2}$$

These numerical errors are:

- The statistical error

$$\epsilon_N(T) = \langle\{f(\overline{\mathbf{Z}}_T)\}_{N,\Delta x}\rangle - \{f(\mathbf{Z}_T)\}_{N,\Delta x} \tag{5.4.3}$$

  which is inherent to all Monte-Carlo methods. The statistical error is random and its asymptotic behaviour is given by the central limit theorem.

- The bias error

$$\epsilon_\infty(T) = \{f(\overline{\mathbf{Z}}_T)\}_{\infty,\Delta x} - \langle\{f(\overline{\mathbf{Z}}_T)\}_{N,\Delta x}\rangle \tag{5.4.4}$$

  The bias, which is a deterministic error, is the difference between the mean value of a quantity for a finite number of particles and the mean value for infinitely many particles, all other parameter being unchanged.

- The time discretization error

$$\epsilon_{\Delta t}(T) = \langle f(\mathbf{Z}_T)\rangle - \langle f(\overline{\mathbf{Z}}_T)\rangle \tag{5.4.5}$$

  where $\langle f(\overline{\mathbf{Z}}_T)\rangle = \{f(\overline{\mathbf{Z}}_T)\}_{\infty,0}$. This deterministic error is due to the numerical integration in time of the stochastic differential system (operator T in Chapter 4).

- The space discretization error

$$\epsilon_{\Delta x}(T) = \langle f(\overline{\mathbf{Z}}_T)\rangle - \{f(\overline{\mathbf{Z}}_T)\}_{\infty,\Delta x} \tag{5.4.6}$$

This deterministic error is due to the exchange of information between the mesh and the particles (A and P operators, and of course F for the fluid, see Chapter 4).

The study of the discretization error has to be performed in a numerical case where the influence of other sources of error is negligible, in other words one has to make sure that the error is almost deterministic. In the following, ideal cases will be chosen where there is no exchange between the mesh and the particles, and therefore no spatial discretization error is present, i.e. $\epsilon_{\Delta x}(T) = 0$. In addition, the ideal system will be linear, which implies the elimination of the bias in the numerical procedure, i.e. $\epsilon_{\infty}(T) = 0$. In such cases, the statistical and the time discretization error can be studied. From Eqs. 5.4.3-5.4.6, for our particular test case we obtain

$$\{f(\overline{\mathbf{Z}}_T)\}_{N,\Delta x} = \{f(\overline{\mathbf{Z}}_T)\}_N = \frac{1}{N}\sum_{i=1}^{N} f(\overline{\mathbf{Z}}_T^i) \tag{5.4.7}$$

where $\overline{\mathbf{Z}}_T^i$ represents the value of $\overline{\mathbf{Z}}_t$ at time $t = T$ for trajectory i. Therefore the statistical error can be written

$$\epsilon_N(T) = \langle f(\overline{\mathbf{Z}}_T)\rangle - \frac{1}{N}\sum_{i=1}^{N} f(\overline{\mathbf{Z}}_T^i) \tag{5.4.8}$$

The central limit theorem tells us that for a sufficiently large number of trajectories, this random variable converges towards a Gaussian random variable $\mathcal{G}$

$$\epsilon_N(T) \xrightarrow[N\to\infty]{} \frac{\sigma[f(\overline{\mathbf{Z}}_T)]}{\sqrt{N}}\mathcal{G} \tag{5.4.9}$$

where $\sigma[f(\overline{\mathbf{Z}}_T)]$ is the r.m.s value of $f(\overline{\mathbf{Z}}_T)$. In summary, in our particular case, the total error is the sum of the statistical and the time discretization error. From previous consideration, one can state that the total error is a random variable which, for a sufficiently large number of trajectories, becomes Gaussian. Its mean and r.m.s. values are given by

$$\langle\epsilon(T)\rangle = \epsilon_{\Delta t}(T)\,, \qquad \sigma[\epsilon(T)] = \frac{\sigma[f(\overline{\mathbf{Z}}_T)]}{\sqrt{N}} \tag{5.4.10}$$

In the simulations, the following approximation is made

$$\langle\epsilon(T)\rangle = \epsilon_{\Delta t}(T) \simeq \{\epsilon(T)\}_M \tag{5.4.11}$$

which means that M simulations with N trajectories each are performed. The previous approximation induces a numerical error $\epsilon_M(T)$ defined by

$$\epsilon_M(T) = \langle \epsilon(T) \rangle - \{\epsilon(T)\}_M \tag{5.4.12}$$

and from the central limit theorem, this random variable converges, for M and N sufficiently large, towards a Gaussian random variable

$$\epsilon_M(T) \xrightarrow[M\to\infty]{} \frac{\sigma[\epsilon(T)]}{\sqrt{M}}\mathcal{G} \xrightarrow[M,N\to\infty]{} \frac{\sigma[f(\mathbf{Z}_T)]}{\sqrt{M\cdot N}}\mathcal{G} \tag{5.4.13}$$

Then we chose a confidence interval and we establish the values of M and N necessary to evaluate the error with a given precision (see Chibbao *et al.*[20]). It is now obvious that a good approximation of the time discretization error can only be obtained when M and N are large and $\sigma[f(\mathbf{Z}_T)]$ is small.
In the following study we will evaluate the error of the second order moments, known them from an analytical solution. So we will have

$$f(Z_T) = x_p^2(T) \qquad f(Z_T) = U_p^2(T) \qquad f(Z_T) = U_s^2(T) \qquad f(Z_T) = U_p(T)U_s(T) \tag{5.4.14}$$

and the error will be computed by

$$\langle \epsilon(T) \rangle = \langle f(\overline{Z}_T) \rangle - \{\{f(\overline{Z}_T)\}_N\}_M \tag{5.4.15}$$

We now look for a system with non-constant coefficients and known analytical solutions.

5.4.1 Analytical solutions

Let us assume that the coefficients depend on time only and look for analytical solutions to the following system

$$\begin{cases} dx_p(t) = U_p\, dt \\ dU_p(t) = -\beta(t)(U_p - U_s)\, dt \\ dU_s(t) = -\alpha(t)U_s\, dt + \sigma(t)dW(t) \end{cases} \tag{5.4.16}$$

The exact solutions to system 5.4.16 are given by

$$\begin{aligned} x_p(t) &= x_p(t_0) + \int_{t_0}^{t} U_p(s)\, ds \\ U_p(t) &= U_p(t_0) exp[-G(t)] + exp[-G(t)] \int_{t_0}^{t} exp[G(s)]\beta(s) U_s(s)\, ds \\ U_s(t) &= U_s(t_0) exp[-F(t)] + exp[-F(t)] \int_{t_0}^{t} exp[F(s)]\sigma(s)\, dW(s) \end{aligned} \tag{5.4.17}$$

where the functions $F(t)$ and $G(t)$ are given by

$$F(t) = \int_{t_0}^{t} \alpha(s)\, ds \qquad \text{and} \qquad G(t) = \int_{t_0}^{t} \beta(s)\, ds \tag{5.4.18}$$

Consequently, the functions $\alpha(t)$, $\beta(t)$ and $\sigma(t)$ must be chosen in such way so that the moments of first and second order, which will be used for the study of the numerical error, can be calculated explicitly. Some constraints must be, however, satisfied so that the system has a physical meaning (positiveness of coefficients, smoothness, etc.). As already suggested by Haworth and Pope [21], in a study of a simpler system, the basic idea is to take for $\alpha(t)$ and $\beta(t)$ functions whose primitives are logarithms. This property eliminates the exponentials. Let us write

$$\alpha(t) = \frac{a}{\alpha_0 t + 1} \qquad \text{and} \qquad \beta(t) = \frac{b}{\alpha_0 t + 1} \tag{5.4.19}$$

where a, b and α_0 take the real positive values and therefore

$$F(t) = [\ln(\alpha_0 s + 1)^k]_{t_0}^{t} \qquad \text{and} G(t) = [\ln(\alpha_0 s + 1)^n]_{t_0}^{t} \tag{5.4.20}$$

where $k = a/\alpha_0$ and $n = b/\alpha_0$. For the diffusion coefficient, an expression which allows control (in time) and exact integration, is proposed, that is

$$\sigma(t) = \sigma_0(\alpha_0 t + 1)^p \tag{5.4.21}$$

where σ_0 takes real positive values and p is real. We set zero initial condition on all variables at time t_0. This implies that all first order moments are equal to zero. In Tables 5.2-5.3 the analytical solution and the computation of second order moments that can be carried out following Eq. 4.4.3 are shown.

Table 5.2: Analytical solutions to system (5.4.16) and analytical expressions for the second order moments of $U_s(t)$ and $U_p(t)$.

By stochastic integration by parts, one has for $U_s(t)$, $U_p(t)$, $x_p(t)$:

$$U_s(t) = \frac{\sigma_0}{(\alpha_0 t+1)^k}\int_{t_0}^{t}(\alpha_0 t+1)^{k+p}\,dW(s) \tag{5.4.22}$$

$$U_p(t) = \frac{\sigma_0 b}{\alpha_0(n-k)}\Big[\frac{1}{(\alpha_0 t+1)^k}\int_{t_0}^{t}(\alpha_0 t+1)^{k+p}\,dW(s) - \frac{1}{(\alpha_0 t+1)^n}\int_{t_0}^{t}(\alpha_0 t+1)^{n+p}\,dW(s)\Big] \tag{5.4.23}$$

$$\begin{aligned} x_p(t) = {} & \frac{\sigma_0 b}{\alpha_0^2(n-k)}\Big[\Big(\frac{1}{k-1}-\frac{1}{n-1}\Big)\int_{t_0}^{t}(\alpha_0 t+1)^{p+1}\,dW(s) \\ & - \frac{(\alpha_0 t+1)^{1-k}}{k-1}\int_{t_0}^{t}(\alpha_0 t+1)^{k+p}\,dW(s) + \frac{(\alpha_0 t+1)^{1-n}}{n-1}\int_{t_0}^{t}(\alpha_0 t+1)^{n+p}\,dW(s)\Big] \end{aligned} \tag{5.4.24}$$

By resorting to Ito's calculus, $\langle U_s^2(t)\rangle$, $\langle U_p^2(t)\rangle$, $\langle U_p(t)U_s(t)\rangle$ can be written:

$$\langle U_s^2(t)\rangle = \frac{\sigma_0^2}{\alpha_0[2(k+p)+1]}\Big[(\alpha_0 t+1)^{2p+1} - \frac{1}{(\alpha_0 t+1)^{2k}}\Big] \tag{5.4.25}$$

$$\begin{aligned} \langle U_p^2(t)\rangle = {} & \frac{1}{\alpha_0}\Big[\frac{\sigma_0 b}{\alpha_0(n-k)}\Big]^2\cdot\Big\{\frac{1}{2(k+p)+1}\cdot \\ & \Big[(\alpha_0 t+1)^{2p+1} - \frac{1}{(\alpha_0 t+1)^{2k}}\Big] - \frac{2}{k+2p+n+1}\Big[(\alpha_0 t+1)^{2p+1} - \frac{1}{(\alpha_0 t+1)^{n+k}}\Big] + \frac{1}{2(n+p)+1}\Big[(\alpha_0 t+1)^{2p+1} - \frac{1}{(\alpha_0 t+1)^{2n}}\Big]\Big\} \end{aligned} \tag{5.4.26}$$

$$\langle U_p(t)U_s(t)\rangle = \frac{\sigma_0^2 b}{\alpha_0^2(n-k)}\Big\{\frac{1}{2(k+p)+1}\Big[(\alpha_0 t+1)^{2p+1} - \frac{1}{(\alpha_0 t+1)^{2k}}\Big] - \frac{1}{k+2p+n+1}\Big[(\alpha_0 t+1)^{2p+1} - \frac{1}{(\alpha_0 t+1)^{n+k}}\Big]\Big\} \tag{5.4.27}$$

which are valid provided that: $n \neq k$, $k \neq 1$, $n \neq 1$, $2(k+p)+1 \neq 0$, $n+2p+k+1 \neq 0$.

Table 5.3: Analytical expressions for the second order moment of $x_p(t)$.

Eq. 5.4.24 is rewritten as $x_p(t) = \Big[\frac{\sigma_0 b}{\alpha_0^2(n-k)}\Big][I_1 + I_2 + I_3]$ and the second order moment becomes:

$$\langle x_p^2(t)\rangle = \Big[\frac{\sigma_0 b}{\alpha_0^2(n-k)}\Big]^2\Big[\sum_{i=1}^{3}\langle I_i^2\rangle + 2\sum_{i=1}^{3}\langle I_i I_j\rangle\Big] \tag{5.4.28}$$

$$\langle I_1^2\rangle = \frac{1}{\alpha_0(2p+3)}\Big[\frac{1}{k-1} - \frac{1}{n-1}\Big]^2[(\alpha_0 t+1)^{2p+3} - 1] \tag{5.4.29}$$

$$\langle I_2^2\rangle = \frac{1}{\alpha_0[2(k+p)+1]}\Big[\frac{1}{k-1}\Big]^2[(\alpha_0 t+1)^{2p+3} - (\alpha_0 t+1)^{2(1-k)}] \tag{5.4.30}$$

$$\langle I_3^2\rangle = \frac{1}{\alpha_0[2(n+p)+1]}\Big[\frac{1}{n-1}\Big]^2[(\alpha_0 t+1)^{2p+3} - (\alpha_0 t+1)^{2(1-n)}] \tag{5.4.31}$$

$$2\langle I_1 I_2\rangle = -\frac{2}{k-1}\Big[\frac{1}{k-1} - \frac{1}{n-1}\Big]\frac{1}{\alpha_0(k+2p+2)}[(\alpha_0 t+1)^{2p+3} - (\alpha_0 t+1)^{1-k}] \tag{5.4.32}$$

$$2\langle I_1 I_3\rangle = \frac{2}{n-1}\Big[\frac{1}{k-1} - \frac{1}{n-1}\Big]\frac{1}{\alpha_0(n+2p+2)}[(\alpha_0 t+1)^{2p+3} - (\alpha_0 t+1)^{1-n}] \tag{5.4.33}$$

$$2\langle I_2 I_3\rangle = -2\frac{1}{k-1}\frac{1}{n-1}\frac{1}{\alpha_0(n+k+2p+1)}[(\alpha_0 t+1)^{2p+3} - (\alpha_0 t+1)^{2-n-k}] \tag{5.4.34}$$

5.4.2 Numerical study

The numerical values of the parameters of system 5.4.16 chosen in the simulation of the general case and the limit cases are displayed in Table 5.4. These values are essentially chosen (i) to respect the constraints of separation of scales in the different cases and (ii) to make sure that the system remains stochastic, that is the diffusion coefficient is of the same order of magnitude of the drift term. For the first order scheme, $N = 3 \cdot 10^4$ trajectories, $M = 100$ simulations, a

	α_0	a	b	σ_0	p
general case	0.5	0.1	0.25	0.5	-1.2
limit case (i)	0.5	0.1	250	0.5	-1.2
limit case (ii)	0.5	200	0.25	50	-1.2
limit case (iii)	0.5	200	250	50	-1.2

Table 5.4: Numerical values of the parameters of system 5.4.16 in the general and limit cases.

final time $T = 2.4s$ and the time steps $\Delta t = [0.4, 0.2, 0.1, 0.05]$ were used. For the second order scheme, of course more trajectories and simulations were necessary, $N = 9 \cdot 10^5$ trajectories, $M = 200$ simulations, a final time $T = 3.2s$ with larger time steps $\Delta t = [0.8, 0.4, 0.2, 0.1]$ were used. The final time for the second order scheme was different from that of the first order scheme only to match an integer number with the largest time step, $\Delta t = 0.8$.

Figures 5.9-5.12 show the results of the simulations in all tested cases, with the error normalized with its maximum value. Fig. 5.9 shows that both schemes have the expected order of convergence in the general case. For limit case (i), Fig. 5.10 shows once again that the expected order of convergence is obtained. When the fluid velocity seen becomes a fast variable, limit case (ii), the second order scheme becomes a first order scheme, see Fig. 5.11. The same phenomenon is observed in limit case (iii), where both fluid velocity seen and particle velocity become fast variables, Fig. 5.12.

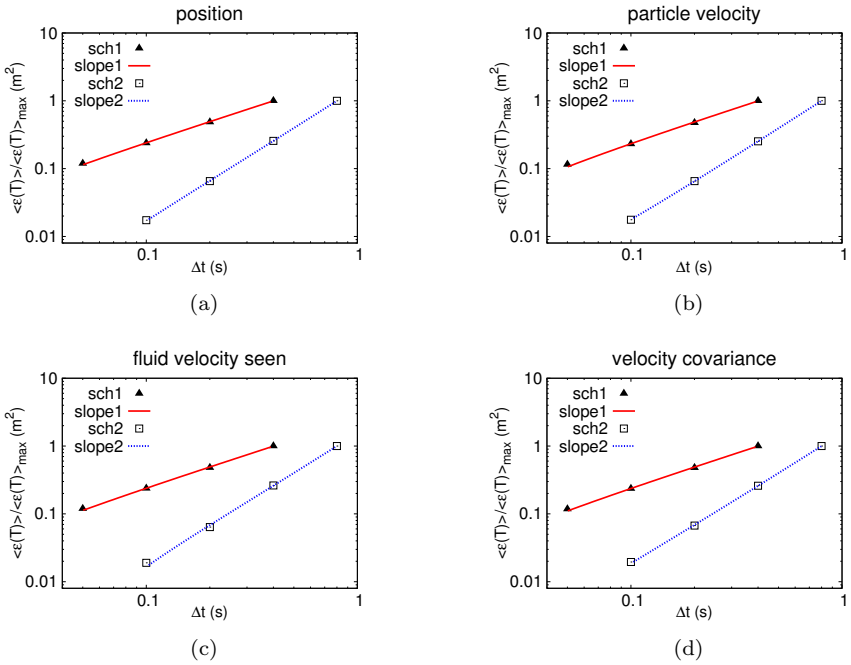


Figure 5.9: Study of the discretization error in the general case: $\alpha_0 = 0.5$, $a = 0.1$, $b = 0.25$, $\sigma_0 = 0.5$, $p = -1.2$. For sch1: $N = 3 \cdot 10^4$, $M = 100$, $T = 2.4s$. For sch2: $N = 9 \cdot 10^5$, $M = 200$, $T = 3.2s$.

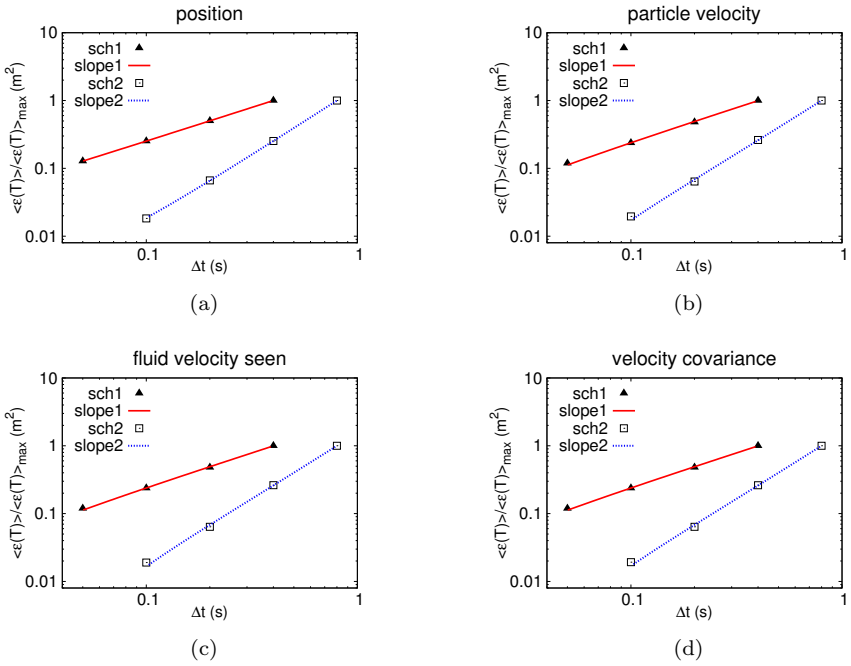


Figure 5.10: Study of the discretization error in the limit case (i): $\alpha_0 = 0.5$, $a = 0.1$, $b = 250$, $\sigma_0 = 0.5$, $p = -1.2$. For sch1: $N = 3 \cdot 10^4$, $M = 100$, $T = 2.4s$. For sch2: $N = 9 \cdot 10^5$, $M = 200$, $T = 3.2s$.

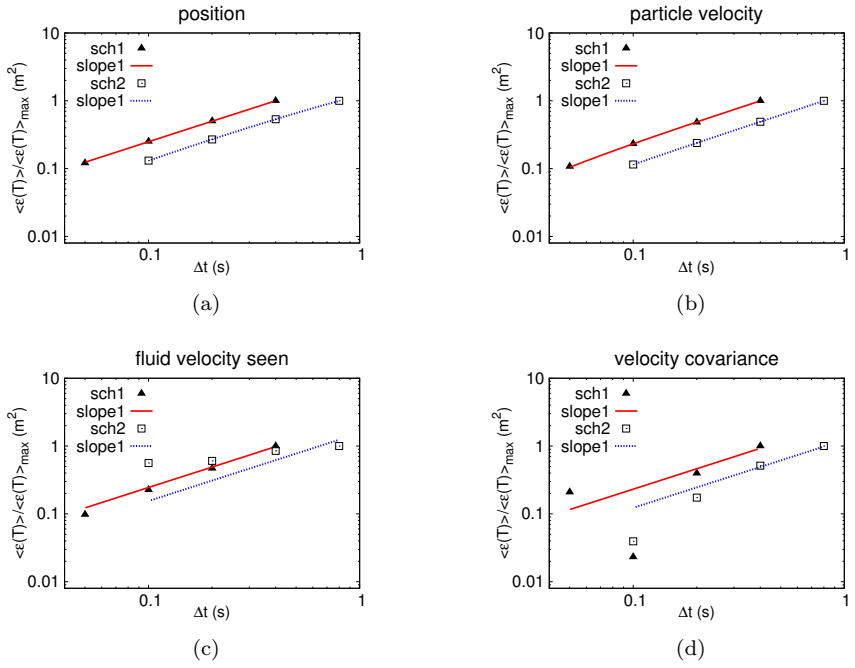


Figure 5.11: Study of the discretization error in the limit case (ii): $\alpha_0 = 0.5$, $a = 200$, $b = 0.25$, $\sigma_0 = 50$, $p = -1.2$. For sch1: $N = 3 \cdot 10^4$, $M = 100$, $T = 2.4s$. For sch2: $N = 9 \cdot 10^5$, $M = 200$, $T = 3.2s$.

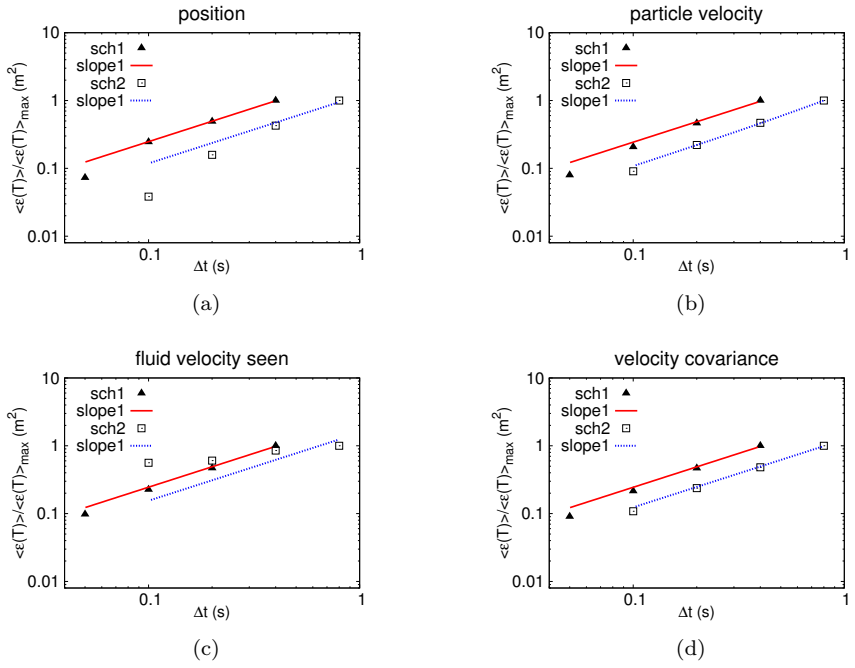


Figure 5.12: Study of the discretization error in the limit case (iii): $\alpha_0 = 0.5$, $a = 200$, $b = 250$, $\sigma_0 = 50$, $p = -1.2$. For sch1: $N = 3 \cdot 10^4$, $M = 100$, $T = 2.4s$. For sch2: $N = 9 \cdot 10^5$, $M = 200$, $T = 3.2s$.

Chapter 6

Results

The flow into which particles are introduced, is a turbulent channel flow of air, assumed incompressible and newtonian. The main fluid physical parameters are summarized in Table 6.2.

Physical details

The reference geometry consists in two infinite flat parallel walls reported in Fig. 4.1, with periodic boundary conditions on the fluid velocity field in stream-wise and span-wise directions, and no slip conditions at the walls. The simulations have been performed on a grid with a coarsening factor with respect to the DNS of $CF = 8$, at the shear Reynolds number $Re_\tau = 300$. When not specified, LES is resolved with no subgrid model.

Channel properties			
Pressure difference	Δp	$(-0.9060, 0, 0)$	$[Pa]$
Domain dimensions	$L_1 = 4\pi h$	251.33	$[mm]$
	$L_2 = 2\pi h$	125.66	$[mm]$
	$L_3 = 2h$	40	$[mm]$
	$L_1^+ = L_1 Re_\tau / h$	3770	$[w.u.]$
	$L_2^+ = L_2 Re_\tau / h$	1885	$[w.u.]$
	$L_3^+ = L_3 Re_\tau / h$	600	$[w.u.]$

Table 6.1: Channel properties.

Fluid properties			
Density	ρ_f	1.3	$[kg/m^3]$
Kinematic viscosity	ν_f	$1.57 \cdot 10^{-5}$	$[m^2/s]$
Shear Reynolds number	Re_τ	300	
Friction velocity	u_τ	0.2355	$[m/s]$

Table 6.2: Fluid properties.

6.1 Test of consistency

We now proceed with some tests which will give a safe basis for future works. In particular we carried out some tests of consistency, to verify if in the limit case where $\tau_p \to 0$ (in previous sections limit case 1), the particles really behave as fluid particles. We are particularly interested in particle distribution, which should remain constant in the whole channel because of the conservation of mass, and in velocity statistics calculated from particle velocity. We call *duplicate* fields all the quantities that are calculated from the LES solver and from particles. Such duplicate quantities should be nearly the same.
To calculate the filtered velocity starting from particle velocity, an averaging operation has to be performed. Thus, we averaged the velocity of all the particles residing within an ensemble domain of characteristic length $\Delta_E = \Delta$, where $\Delta = \sqrt[3]{\Delta_x \Delta_y \Delta_z}$, centered on the same cell of the LES discretization exactly in the same manner as we did in the integration of the SDEs.
First of all the sensitivity of the model to the interpolation technique has been studied. We proposed three types of approach

- *no-interpolation*: for each particle we identify the cell in which it is residing and we assign to the filtered quantities appearing in the model equation, the value calculated from the LES solver at one node of the cell (so, no interpolation is done). This type of zeroth order projection is also not symmetric

- *NGP-interpolation* (nearest grid point): it is still a zeroth order interpolation, but instead of the value of one node it is assigned the averaging of all the nodes of the identified cell. This one has the advantage that it is still very simple and symmetric

- *interpolation*: we interpolated the LES quantities at particles position with a first order interpolation technique

In the following, all quantities with an overbar are Reynolds averaged quantities, which means that an averaging operation over homogeneous directions and in time has been made. We also

recall that $\overline{\langle u^2 \rangle}_L$ indicates the root mean square of the filtered velocity, thus

$$\overline{\langle u^2 \rangle}_L = \sqrt{\overline{(\langle U \rangle_L - \overline{\langle U \rangle}_L)^2}} \tag{6.1.1}$$

while $\tau_L(u,u)$ is a subgrid stress, i.e.

$$\tau_L(u,u) = \frac{1}{N} \sum_{i=1}^{N} (U_s - \langle U_s \rangle_L) \times (U_s - \langle U_s \rangle_L) \tag{6.1.2}$$

where N is the number of particles in the cell considered.

Focusing in particular on particles distribution in the domain we see from Figure 6.1 that the

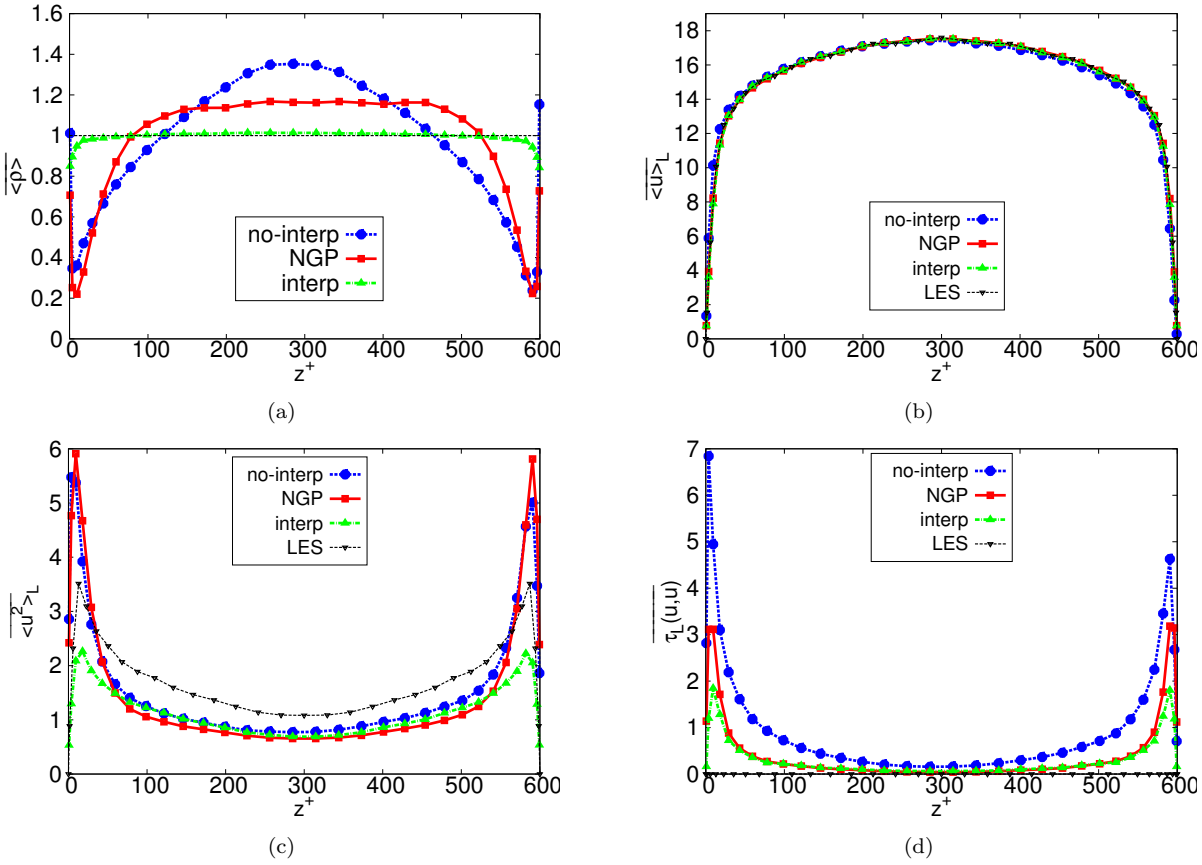


Figure 6.1: Reynolds averaged quantities ($\Delta t^+ = 3000$) obtained with different particle model: no-interpolation of the LES and particles quantities, NGP and full interpolation of first order. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES. (a) particle density, (b) filtered streamwise velocity, (c) filtered streamwise velocity rms, (d) xx component of the subgrid stress tensor.

only suitable model is the one with interpolation, because of particle spurious concentration in the other cases. Even in the case with interpolation there is a lack of particle near the wall of the order of 10% which is not fully satisfactory but acceptable (in particular if we think that we are interested in inertial particles and not fluid particles). It is even less worrying if we plot the particle density normalized with particle position at beginning of the time interval used for averaging, Fig. 6.2, from which we can see that the spurious drift occurs in the transient initial phase. Also the other fluid statistics show a better behaviour in the case of interpolation. The velocity root mean square is clearly underestimated with interpolation with respect to the LES one. It is interesting to compare the results also with the DNS case. The results are much more closer, see Fig. 6.3. Instantaneous correlation between LES and VFDF fields are quite good as we can see in Fig. 6.4. A final remark to make is on subgrid stresses. They are higher for the stochastic model but it is worth remembering that they have a different status.

From now on we will focus for further tests only on the model with interpolation. Some tests

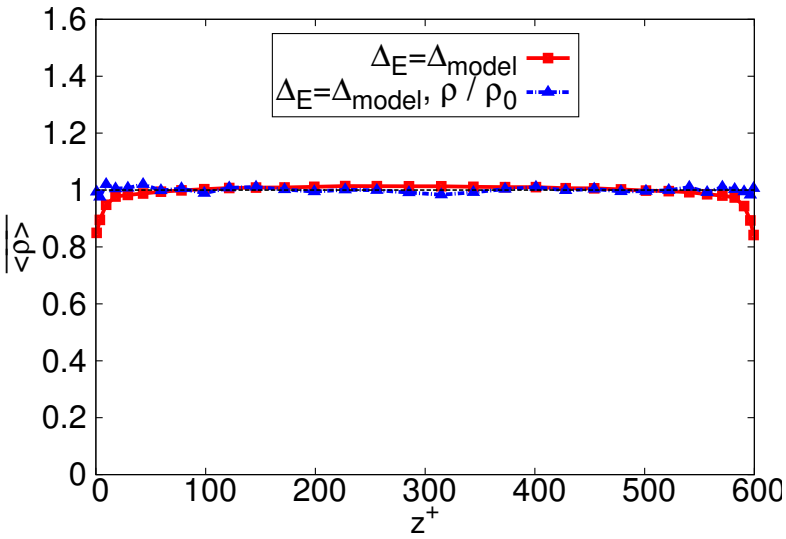
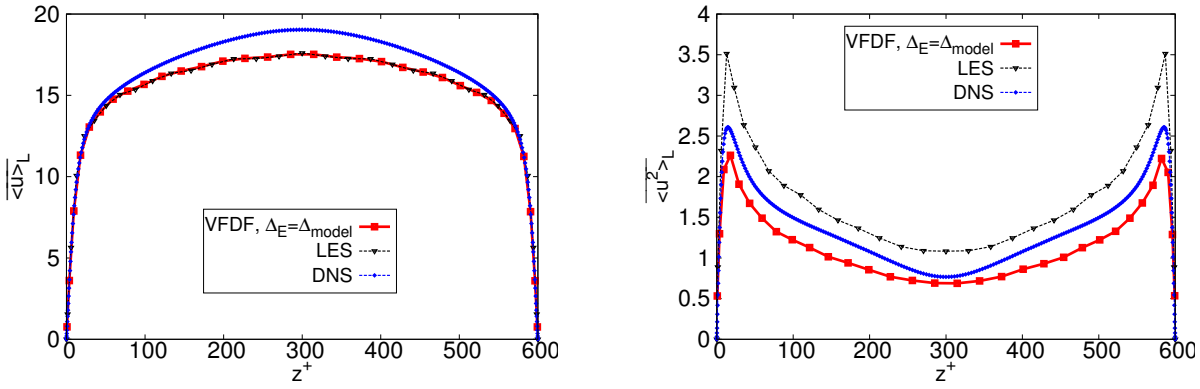


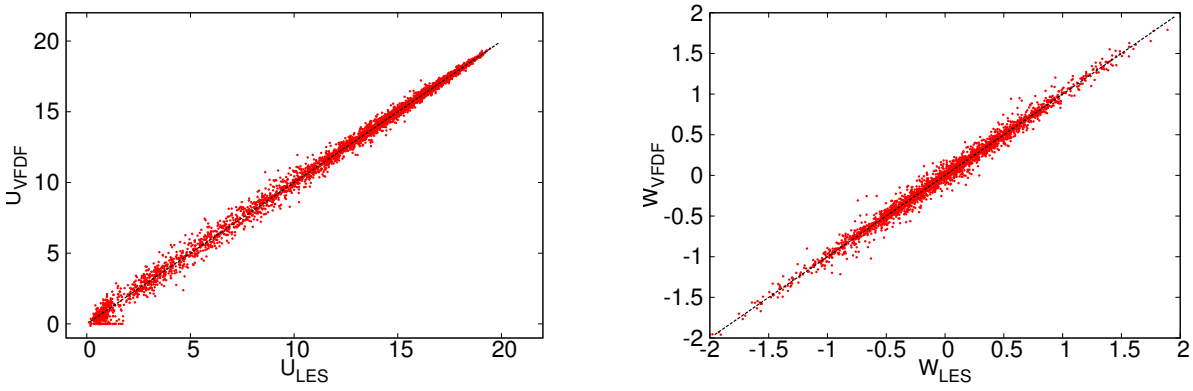
Figure 6.2: Reynolds averaged density ($\Delta t^+ = 3000$) normalized with particle position at the beginning of the time interval used to averaging. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES.

are then made to evaluate the importance of some terms in the model equations. In particular we focused our attention on the filtered velocity in the mean drift term, see Eq. 3.2.1. We studied the differences in interpolating it at particle position or not, Fig. 6.5. Further tests are made to see the importance of the term of pressure gradient, Fig. 6.5. As we can see from Figure 6.5 it



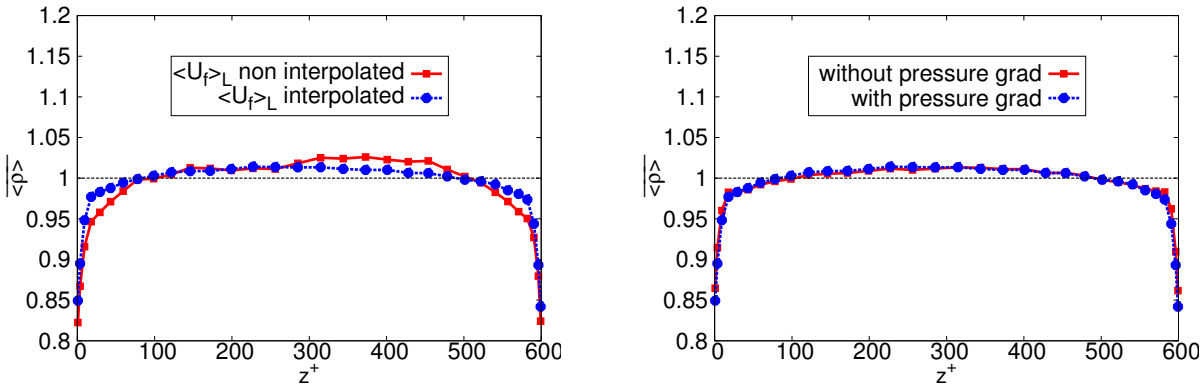
(a) (b)

Figure 6.3: Reynolds averaged quantities ($\Delta t^+ = 3000$) plotted against LES and DNS results. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES. (a) filtered stream-wise velocity, (b) filtered stream-wise velocity rms.



(a) (b)

Figure 6.4: Instantaneous velocity correlation. Scatter plot of velocity evaluated from LES and from VFDF at cell center: (a) stream-wise component, (b) wall normal component. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES.



(a) (b)

Figure 6.5: Reynolds averaged density ($\Delta t^+ = 3000$) obtained in different test case: (a) test on $\langle U_f \rangle_L$, (b) test on pressure gradient. We consider a mean number of particles per cell of $N = 40$, initially random distributed over the entire channel. The grid used for the evaluation of the statistics is fixed and it is the same used for LES.

seems that the pressure gradient has a negligible effect in this limit case. Further statistics are not reported for these two test cases because there are not significant differences between them and the reference case.

For reliable statistics with minimal numerical dispersion, it is desirable to minimize the size

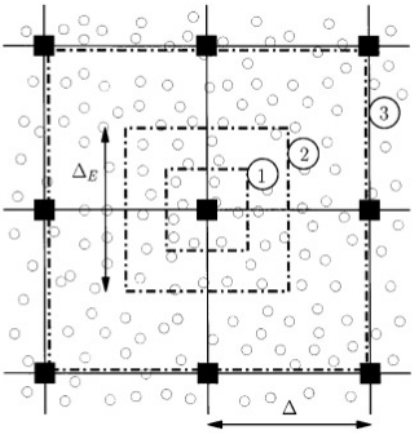


Figure 6.6: Grid used for evaluating statistics with variable cell length: Δ_E is the length of the new grid centered on the nodes of the previous one, Δ is the length of the previous grid.

of ensemble domain and maximize the number of the Monte Carlo particles. In this way the

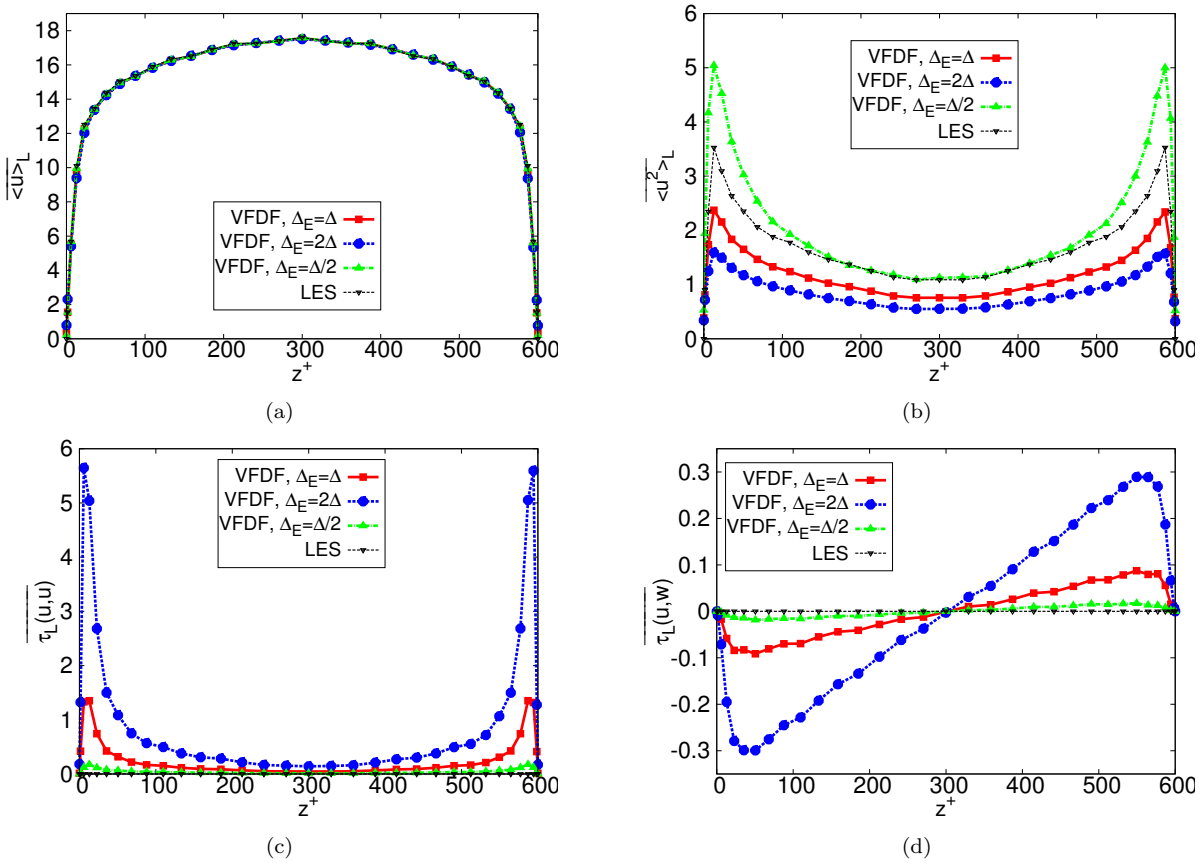


Figure 6.7: Reynolds averaged quantities ($\Delta t^+ = 3000$, $N_E = 40$) obtained for different cell length. (a) filtered streamwise velocity, (b) filtered streamwise velocity rms, (c) xx component of the subgrid stress tensor, (d) xz component of the subgrid stress tensor

ensemble statistics would tend to the desired filtered values

$$\begin{aligned} \langle a \rangle_E &= \frac{1}{N_E} \sum_{n \in \Delta_E} a^{(n)} \xrightarrow[\Delta_E \to 0]{N_E \to \infty} \langle a \rangle \\ \tau_E(a,b) &= \frac{1}{N_E} \sum_{n \in \Delta_E} (a^{(n)} - \langle a \rangle_E)(b^{(n)} - \langle b \rangle_E) \xrightarrow[\Delta_E \to 0]{N_E \to \infty} \tau(a,b) \end{aligned} \tag{6.1.3}$$

where $a^{(n)}$, $b^{(n)}$ denotes the information carried by the nth particle. So, we also carried out a study in which we varied the length of the cell adopted to calculate the filtered values, and a study in which we fixed the cell length and we varied the number of particles per cell. In the first study (and only in that one) we used a different grid with respect to the one used in LES. It is a staggered grid centered on the nodes of the LES one, see Fig. 6.6. From results in Fig. 6.7 we notice that, according to our expectations, the root mean square of the filtered velocity increases

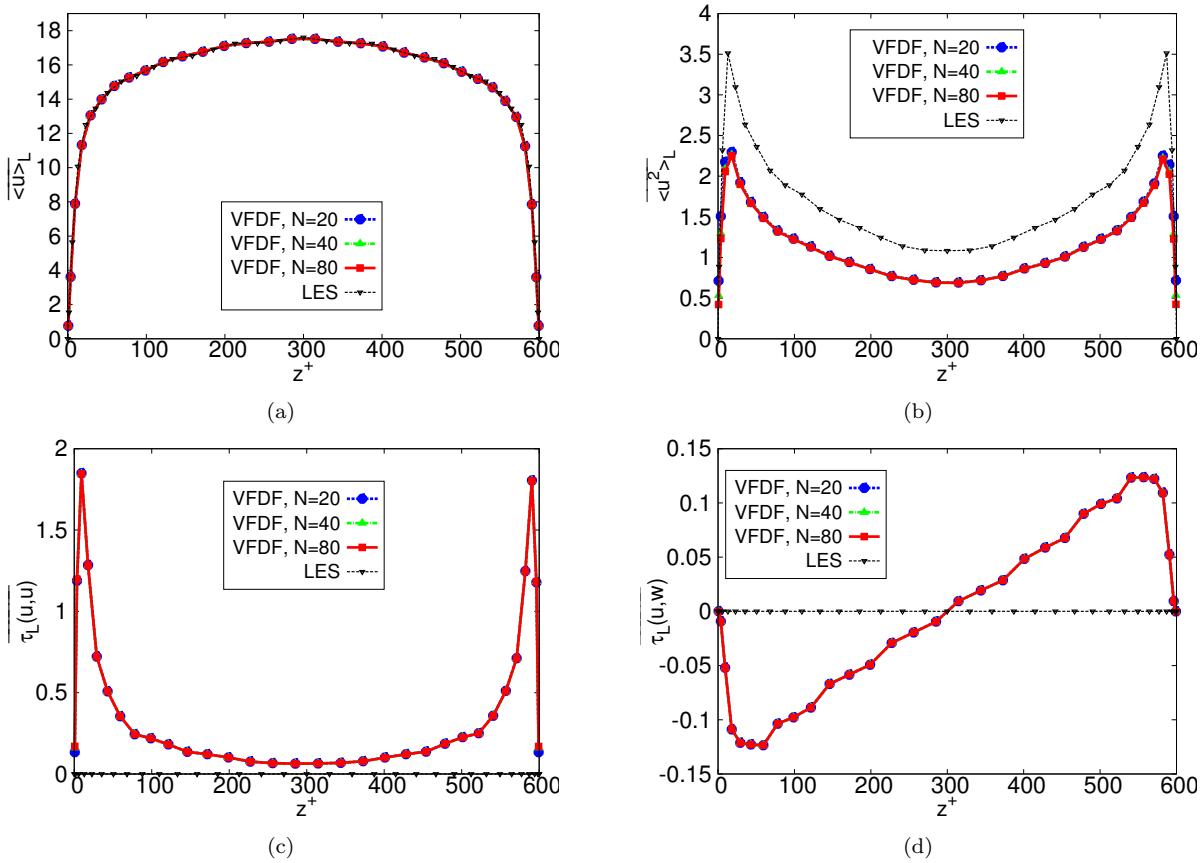


Figure 6.8: Reynolds averaged quantities ($\Delta t^+ = 3000$, $\Delta_E = \Delta$) obtained for different number of particles per cell. (a) filtered streamwise velocity, (b) filtered streamwise velocity rms, (c) xx component of the subgrid stress tensor, (d) xz component of the subgrid stress tensor.

reducing the cell length, while the subgrid stresses decrease. Instead, we can notice from Fig. 6.8 that in the evaluation of averaged statistics the number of particles per cell does not affect so much the results. This can be explained with the central limit theorem which states that the statistical error decreases with the inverse of the square root of N, thus with an increasing factor of 2 on the number of particles per cell, we obtain only a reduction of the error of the order $1/\sqrt{2}$. Since we are averaging in homogeneous directions and in time over 2000 time steps, this has the same effect of increasing particles number when we are evaluating the statistics and so increasing the number of particles has a negligible effect with respect to this operation, except for the pure statistical error on instantaneous quantities (because the number of particles in a cell affects the averaging to evaluate the filtered quantities that are necessary in time advancement, see Eq. 3.2.1).

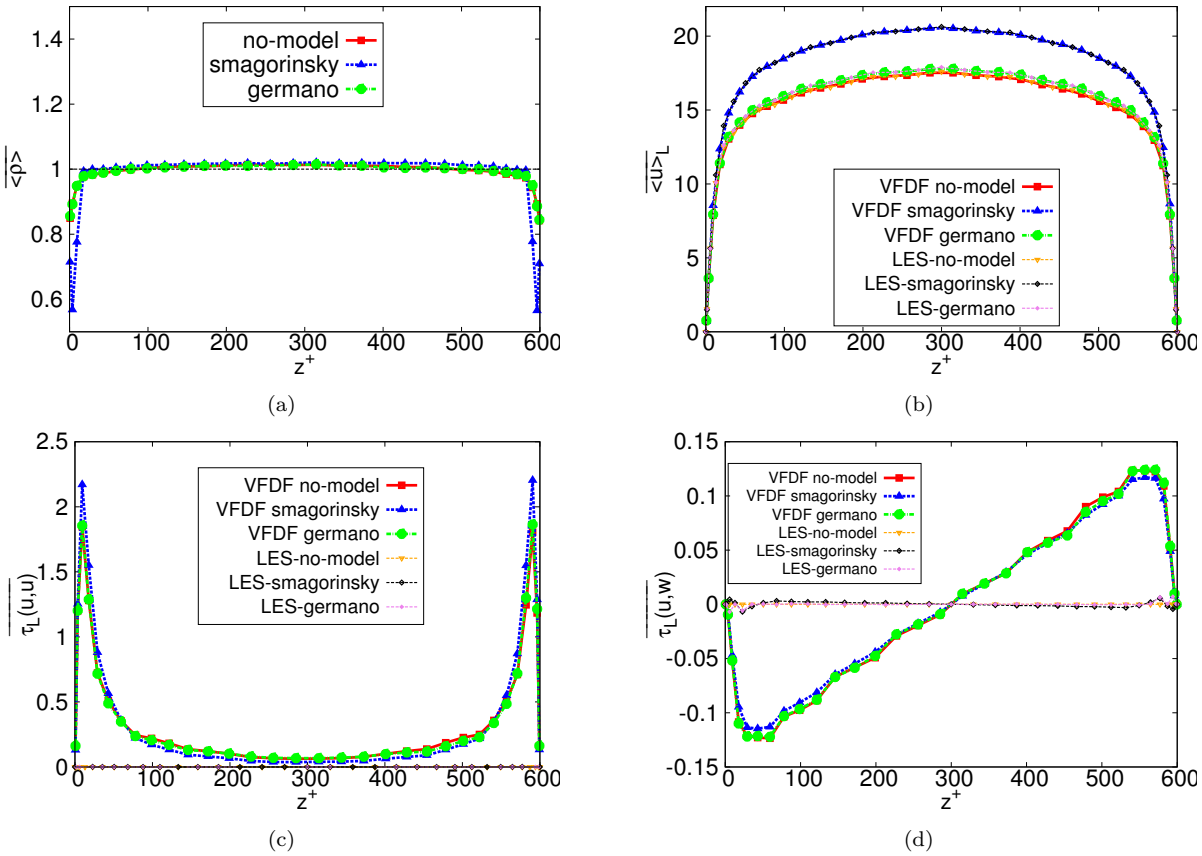
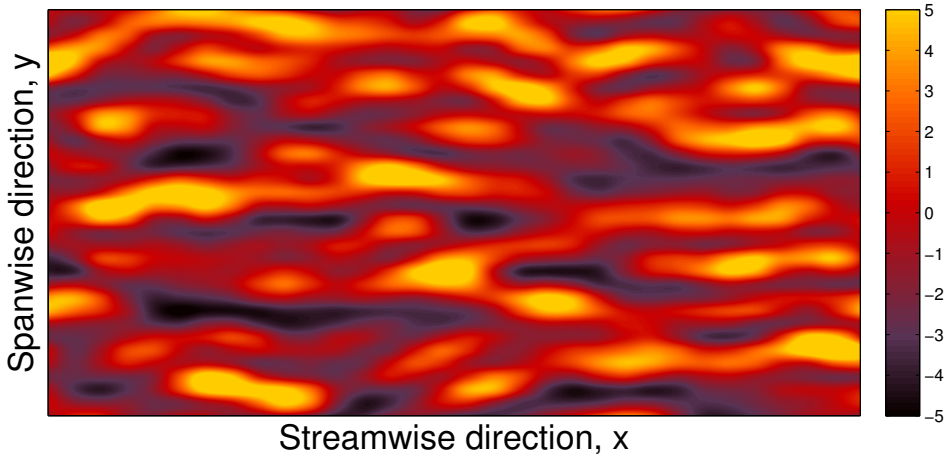
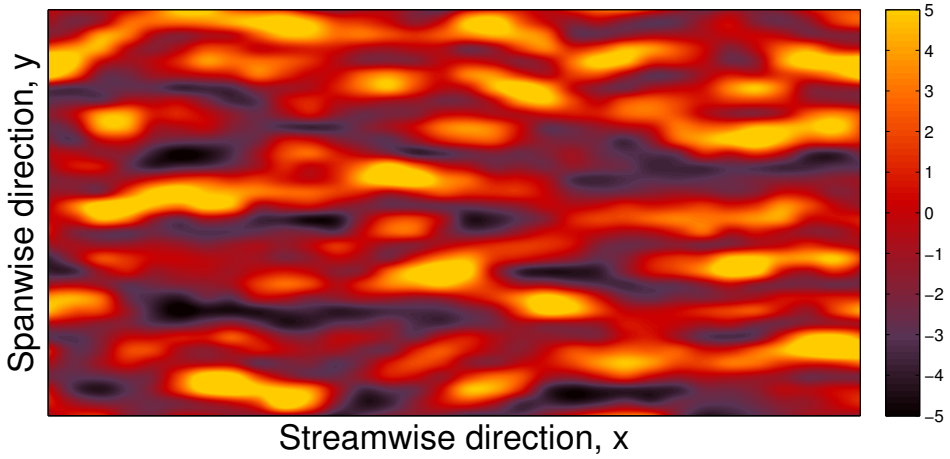


Figure 6.9: Reynolds averaged quantities ($\Delta t^+ = 3000$, $N = 40$) obtained with different fluid subgrid models for LES: no-model, smagorinsky and germano. (a) particle density, (b) filtered stream-wise velocity, (c) xx component of the subgrid stress tensor, (d) xz component of the subgrid stress tensor.

Further tests are carried out to see the importance of the model used for subgrid stresses in LES. Fig .6.9 reports the result for three different case: no model, Smagorinsky and Germano. This test is necessary since the stochastic VFDF model is not perfectly consistent with none of them when coupled in a hybrid algorithm like our. So it is useful to see which one suites better the stochastic model and gives the better results. An evident result is that coupling with Smagorinsky model is quite worse than the other two in predicting particles density over the channel. The other statistics on velocity are instead very similar in each of the three cases. We still notice high subgrid stresses when evaluated from VFDF, but as explained above it is not so meaningful. Fig. 6.10 shows qualitatively agreement between velocity fluctuations from LES and from VFDF, as was already shown from Fig. 6.4. We can also see that in the VFDF case the entity of fluctuations is a bit smaller than in the LES case, which is in agreement with the



(a)



(b)

Figure 6.10: Fluid streaks in the near wall region. Low-speed (resp. high-speed) streaks are rendered using colored contours of negative (resp. positive) streamwise fluctuating velocity u'_x on a horizontal plane at $z^+ = 10$ from the wall. Panel (a) shows the streaky structures obtained from the LES; panel (b) shows the streaky structures from VFDF. LES is computed without subgrid model, and a number of particle per cell $N = 40$ is adopted.

lower values of the velocity root mean square in Fig. 6.3.

6.2 Test with inertial particles

6.2.1 Simulation details

Physical details

Flow properties are the same enunciated in the previous section. From the data reported in Tables 6.2-6.3 for the physical details, the parameters of the particles tracked are summed up in Table 6.4 and obtained from the Stokes number as follows:

$$\tau_p = \frac{\rho_p d_p^2}{18\mu_f}\,, \qquad \tau_f = \frac{\nu}{u_\tau^2} \tag{6.2.1}$$

$$St = \tau_p^+ = \frac{\tau_p}{\tau_f} = \Big(\frac{d_p Re}{H}\Big)^2 \frac{\rho_p}{18\rho_f} \tag{6.2.2}$$

$$d_p = \frac{H}{Re}\sqrt{\frac{18\rho_f St}{\rho_p}} \tag{6.2.3}$$

$$d_p^+ = d_p \frac{Re}{H} = \sqrt{\frac{18\rho_f St}{\rho_p}} \tag{6.2.4}$$

Particle properties			
Density	ρ_p	1000	$[kg/m^3]$
Stokes number	St	$1, 5, 25$	
Number of particles	N	1310720	

Table 6.3: Particle properties.

St	τ_p $[s]$	d_p^+ $[w.u.]$	d_p $[\mu m]$
1	$0.283 \cdot 10^{-3}$	0.153	1
5	$1.415 \cdot 10^{-3}$	0.342	22.8
25	$7.077 \cdot 10^{-3}$	0.763	50.9

Table 6.4: Particle parameters.

Numerical details

For the LES solver we used two different grids as reported in Tab. 6.5, one with a coarsening factor of 8 in all three directions with respect to the DNS grid, and the other with a coarsening

factor of 8 in the homogeneous directions and of 4 in wall-normal direction. We remark that with such a coarse LES grid, we are moving away from DNS, which means a very severe test case. Particles are set into the channel randomly in a flow fully developed and concentrations of particles in wall normal direction are computed at different times and compared in the three different case: DNS with particle tracking, LES with particle tracking with no subgrid model for particles, LES with stochastic particle tracking. The parameters of the simulations are reported in Tab. 6.5. DNS results have been picked by a database of the University of Udine. As we are concerned with a Monte-Carlo simulation, a large number of particles is requested to ensure a small statistical error, so we have chosen the same number of particles used in the fluid test case. We found that there was no significance difference between a number of particles per cell of $N_{pc} = 40$ and $N_{pc} = 80$. Thus we chose $N_{pc} = 40$ which gives a total number of particle in the domain of $N = 1310720$ with a 32^3 grid.

Simulation parameters			
Time step	Δt	$5 \cdot 10^{-4}$	$[s]$
	$\Delta t^+ = \Delta t u_\tau^2/\nu_f$	1.766	$[w.u.]$
Time computed	$\Delta t_1 = t_1 - t_{START}$	0.6029	$[s]$
	$\Delta t_1^+ = t_1^+ - t_{START}^+$	2130	$[w.u.]$
	$\Delta t_2 = t_2 - t_{START}$	1.223	$[s]$
	$\Delta t_1^+ = t_1^+ - t_{START}^+$	4320	$[w.u.]$
DNS grid dimension	$n_1 \times n_2 \times n_3$	$256 \times 256 \times 257$	
$CF8$ grid dimension	$n_1 \times n_2 \times n_3$	$32 \times 32 \times 33$	
$CF8_4$ grid dimension	$n_1 \times n_2 \times n_3$	$32 \times 32 \times 65$	

Table 6.5: Simulation parameters.

6.2.2 Results

We limit ourselves in this work to observe results concerning particle preferential concentration because it is our final purpose. First of all we report particle concentration for the $CF8$ grid on which we performed a LES with no subgrid model coupled with our stochastic particle tracker. The results shown in Fig. 6.11 are not satisfactory at all because are even worse than that with no subgrid model for particles. This led us to modify the diffusion term in the stochastic equation for U_s. In fact, following Minier and Peirano [16], we can see that they adopt a slightly different diffusion term (the difference is the presence of $\tilde{k}/k$) of the form

$$B_{s,ij} = \sqrt{\langle\epsilon\rangle \left(C_0 b_i \tilde{k}/k + \frac{2}{3}(b_i \tilde{k}/k - 1) \right)} \tag{6.2.5}$$

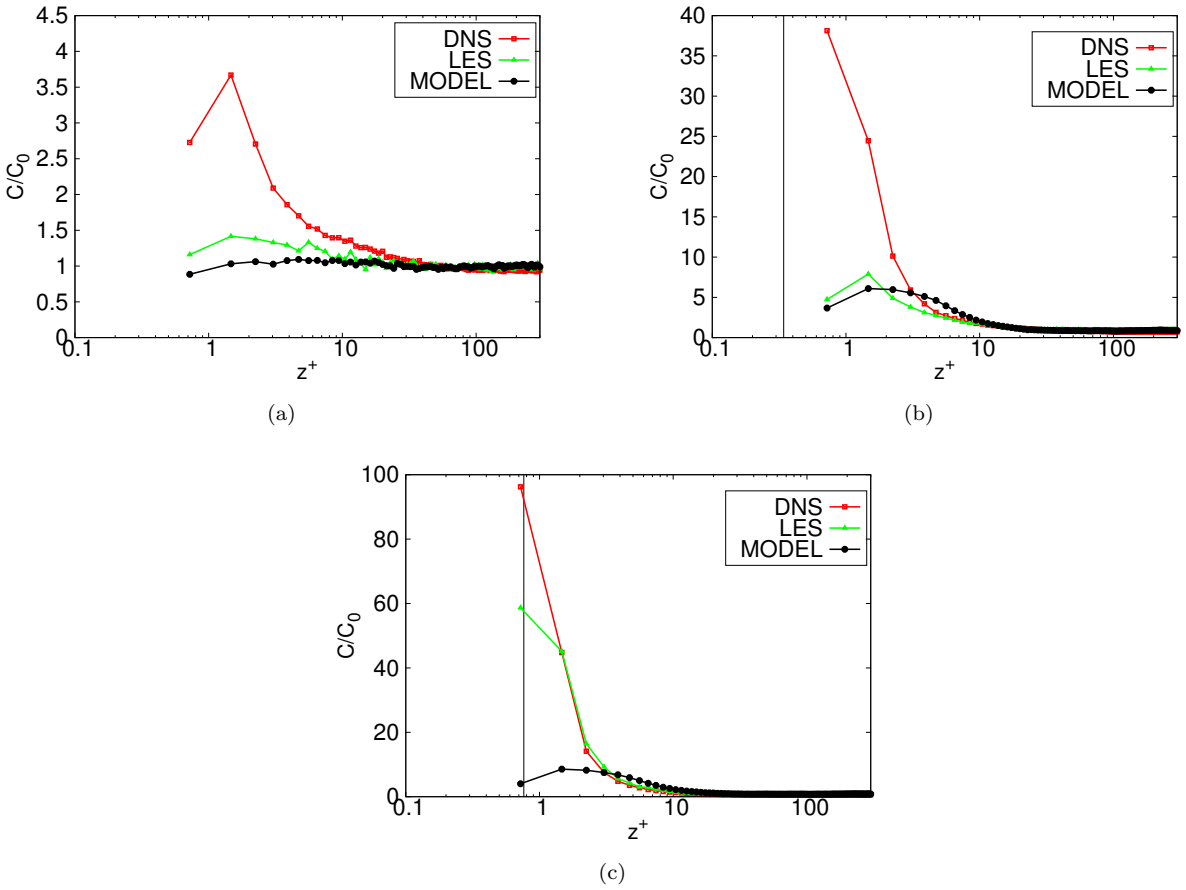


Figure 6.11: Particle concentration obtained with DNS (DNS), LES without SGS modeling in the particle equation of motion (LES) and LES with stochastic model for particles (MODEL). Tests on a coarse computational grid of $32 \times 32 \times 33$ grid nodes ($CF8$). In panel (a) $St = 1$ particles, in panel (b) $St = 5$ particles and in panel (c) $St = 25$ particles. Concentrations are computed at $\Delta t_1^+ = t_1^+ - t_{START}^+ = 2130$ $[w.u.]$.

with

$$\tilde{k} = \frac{3}{2}\frac{\sum_{i=1}^{3} b_i \langle u_i^2 \rangle}{\sum_{i=1}^{3} b_i} \tag{6.2.6}$$

This term is developed in the RANS context, but adopting the rules of stochastic calculus and of filters, we can readapt it to the LES case. If we follow the same rules adopted by Minier and Peirano, we can construct that term in three steps:

- We first consider the simplest case of stationary isotropic turbulence, when mean velocities can be taken as zero. The stationarity constraint, $d\langle U_s^2 \rangle_L / dt = 0$, implies that $B_s^2 = 2\langle U_s^2 \rangle_L / T_L^*$. Then by substituting the expression for $T_L^* = T_L / b$, we find

$$B_s = \sqrt{C_0 b \epsilon_r} \tag{6.2.7}$$

- For the case of fluid velocity seen stationary but not necessarily isotropic, the above closure of the diffusion term is not satisfactory. Due to the anisotropy, the drift and diffusion terms do not balance to ensure that the stationarity constraint is respected. To resolve this difficulty a modified SGS kinetic energy is introduced

$$\tilde{k}_{SGS} = \frac{3}{2}\frac{\sum_{i=1}^{3} b_i [\langle U_{s,i}^2 \rangle_L - \langle U_{s,i} \langle U_{s,i} \rangle_L \rangle_L]}{\sum_{i=1}^{3} b_i} \tag{6.2.8}$$

  Using the modified SGS kinetic energy of the fluid velocity seen, the diffusion term is modified

$$B_{s,i} = \sqrt{C_0 b_i \epsilon_r \frac{\tilde{k}_{SGS}}{k_{SGS}}} \tag{6.2.9}$$

- The next step is to consider the general case of a non-stationary turbulence. A new constraint has to be taken into account, $d\langle U_s^2 \rangle_L / dt = -2\epsilon_r$. Thanks to stochastic calculus and filtering rules we obtain for the diffusion term

$$B_{s,i} = \sqrt{\epsilon_r \Big(C_0 b_i \frac{\tilde{k}_{SGS}}{k_{SGS}} + \frac{2}{3}\Big(b_i \frac{\tilde{k}_{SGS}}{k_{SGS}} - 1 \Big) \Big)} \tag{6.2.10}$$

  with also the expression of the time scal T_L^* slightly modified

$$T_L^* = \Big(\frac{1}{2} + \frac{3}{4} C_0 \Big)^{-1} \frac{k_{SGS}}{\epsilon_r} \tag{6.2.11}$$

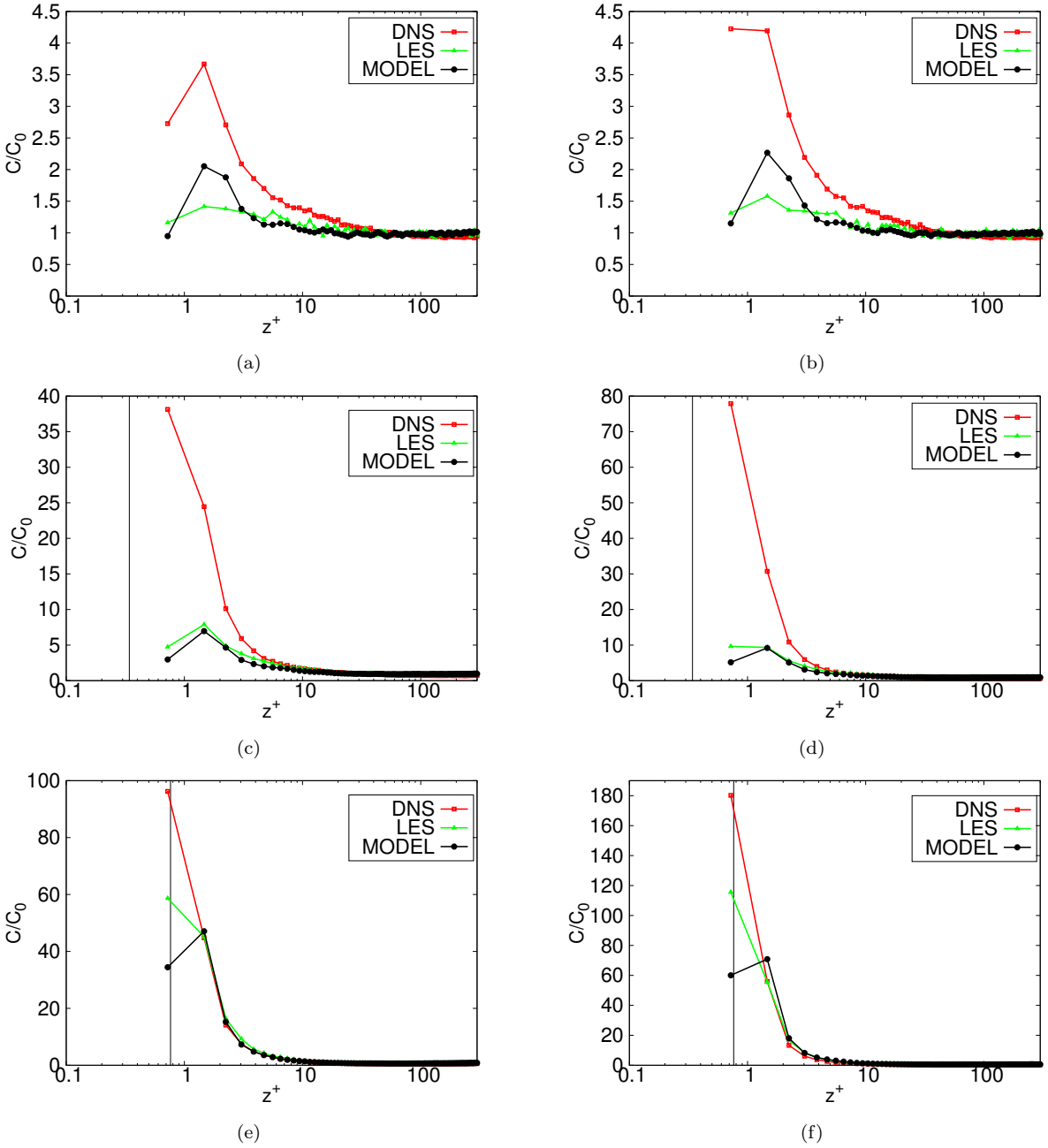


Figure 6.12: Particle concentration obtained with DNS (DNS), LES without SGS modeling in the particle equation of motion (LES) and LES with stochastic model for particles (MODEL). Tests on a coarse computational grid of $32 \times 32 \times 33$ grid nodes ($CF8$). In panels (a)-(b) $St = 1$ particles, in panel (c)-(d) $St = 5$ particles and in panel (e)-(f) $St = 25$ particles. Concentrations are computed at $\Delta t_1^+ = t_1^+ - t_{START}^+ = 2130$ $[w.u.]$ on left panels and at $\Delta t_2^+ = t_2^+ - t_{START}^+ = 4320$ $[w.u.]$ on right panels.

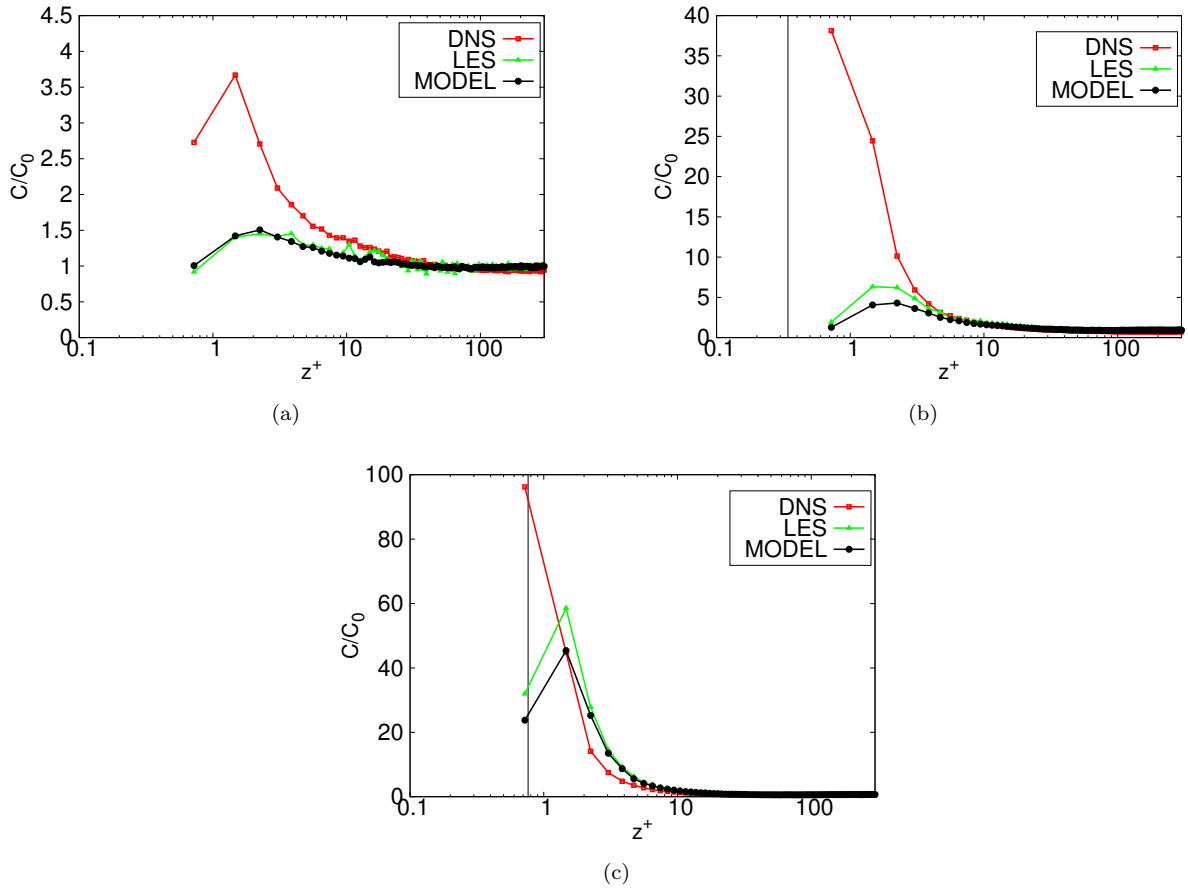


Figure 6.13: Particle concentration obtained with DNS (DNS), LES without SGS modeling in the particle equation of motion (LES) and LES with stochastic model for particles (MODEL). Tests on a coarse computational grid of $32 \times 32 \times 65$ grid nodes ($CF8_4$). In panel (a) $St = 1$ particles, in panel (b) $St = 5$ particles and in panel (c) $St = 25$ particles. Concentrations are computed at $\Delta t_1^+ = t_1^+ - t_{START}^+ = 2130$ $[w.u.]$.

The new diffusion term improves significantly the accuracy of concentrations, taking it back to the level of the LES with no subgrid model for particles, see Fig. 6.12 . There is still some trouble with the first point near the wall where the concentration is always underestimated, but we can see that for low Stokes number the model behaves quite well and concentration profiles are improved with respect to the LES case with no subgrid model for particles.
For completeness we show also the results obtained with the $CF8_4$ grid in Fig. 6.13 . Unfortunately we do not see significant improvements, but we underline that even this grid spacing is very coarse for a $Re_\tau = 300$, and that usually the number of grid nodes is not reduced in the wall-normal direction, if compared with DNS, while we decided to do it for a more severe test.
Finally we tried to achieve some improvement modifying slightly the way we computed dissipation. First of all we add, in the computation of dissipation through Smagorinsky formula, the Van Driest factor (wall function) which should improve the behaviour near the wall. Secondly we changed procedure, calculating dissipation from U_s instead that from the LES flow field. In fact, up to this point we are calculating dissipation through the Smagorinsky formula using in it the LES flow field for velocity derivatives and then we evaluate subgrid kinetic energy from relations

$$\epsilon_r = -\tau_{ij}\frac{\partial\langle U_i\rangle_L}{\partial x_j} = (C_S\Delta)^2|\mathcal{S}|^3 \tag{6.2.12}$$

and

$$k_{SGS} = C_\epsilon(\Delta\epsilon_r)^{2/3} \tag{6.2.13}$$

We now invert the procedure calculating first the SGS kinetic energy from U_s

$$k_{SGS} = \frac{1}{2}\sum_{i=1}^{3}\tau_{ii} = \frac{1}{2}\sum_{i=1}^{3}[\langle U_{s,i}^2\rangle_L - \langle U_{s,i}\rangle_L\langle U_{s,i}\rangle_L] \tag{6.2.14}$$

then we evaluate dissipation by

$$\epsilon_r = \Big(\frac{k_{SGS}}{C_\epsilon}\Big)^{3/2}\frac{1}{\Delta} \tag{6.2.15}$$

We found in this case a better behaviour for all Stokes number and in particular for small particles the concentration profiles are very close to the DNS ones.

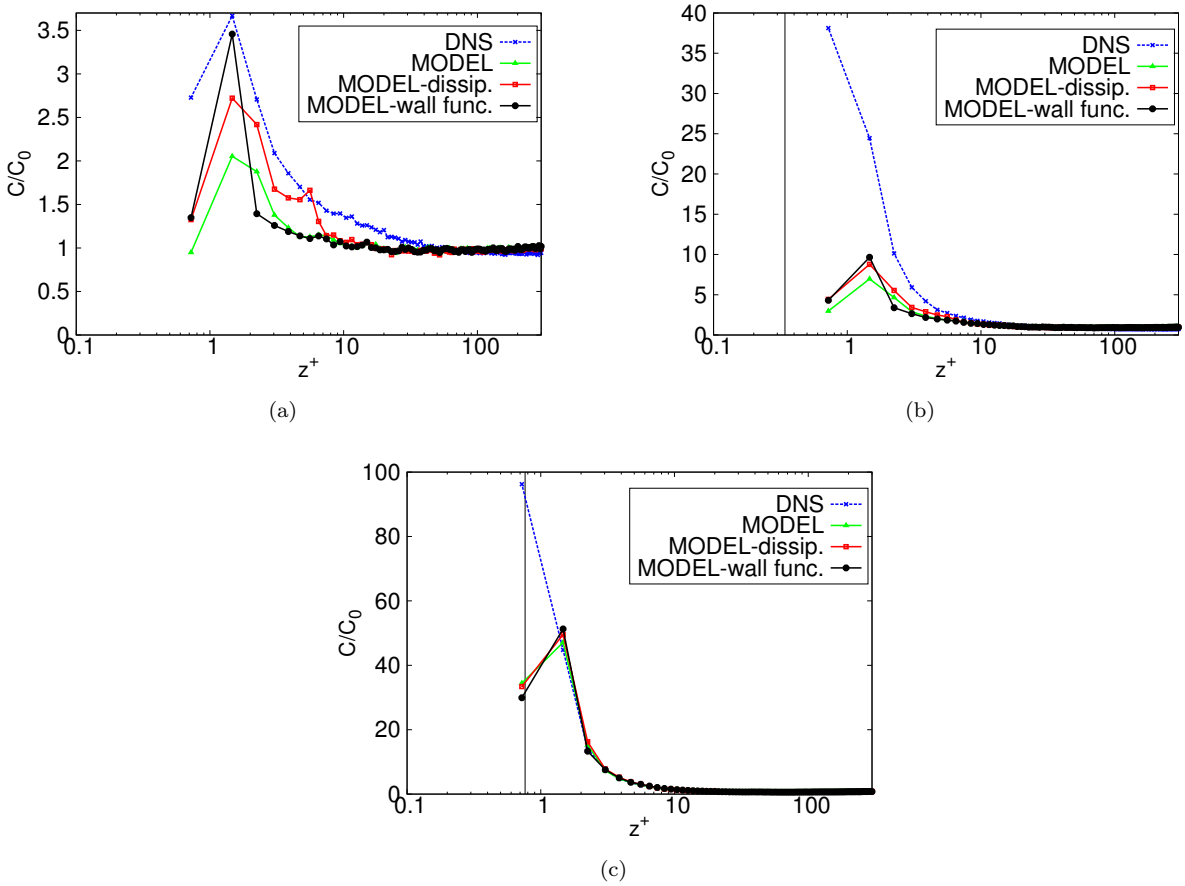


Figure 6.14: Particle concentration obtained with DNS, LES with stochastic model for particles (MODEL), LES with stochastic model for particles and wall function for dissipation (MODEL-wall func.) and LES with stochastic model for particles and new computed dissipation (MODEL-dissip.). Tests on a coarse computational grid of $32 \times 32 \times 33$ grid nodes ($CF8$). In panel (a) $St = 1$ particles, in panel (b) $St = 5$ particles and in panel (c) $St = 25$ particles. Concentrations are computed at $\Delta t_1^+ = t_1^+ - t_{START}^+ = 2130$ $[w.u.]$.

Chapter 7

Conclusions

We now recall what has been done throughout this work and focusing the attention on the interesting results achieved. After the work of implementation a mathematical validation has been carried out, showing a correct agreement between numerical results and analytical solution, which made us sure that there were not errors in the simulation of Gaussian vectors, see Sec. 4.4. We obtained this result for the general case and for all limit cases discussed in Sec. 3.2. Moreover we analysed the time discretization error, finding the expected accuracy respectively for the first and the second order schemes in time. We found also that when the fluid velocity seen becomes a fast variable, the second order scheme has a first order accuracy, as expected from analytical solution of the limit systems.

After this first part of mathematical validation, we carried out physical tests, starting with the fluid limit case, i.e. $\tau_p \to 0$. We required, in this limit case, particles to behave as fluid particles, so that fields extracted from particle quantities should be the same as those calculated from LES. We compared density, which should remain constant for the conservation of mass, and other statistical moments (mean velocity, velocity root mean square, subgrid stresses tensor components). Results showed that interpolation is a must, mainly because with other simpler techniques the constraint of constant density was not satisfied. Mean velocity is always in perfect agreement with the one calculated from LES, while velocity root mean square of the filtered field is always lower, but closer to the DNS one. We showed also that, as for statistical quantities, there is a close similarity also for instantaneous fields, i.e. there is a strong correlation between LES velocity field and VFDF velocity field. Further tests have been made on sensitivity to the number of particle per cell and to the grid spacing in evaluating statistics. We found that there

was not much sensitivity to the number of particles, while reducing grid spacing (the ideal case would be $\Delta_E \to 0$) a convergence towards LES quantities is reached, especially for subgrid stress tensor components. Finally we made a sensitivity test to the subgrid model used in LES and we found that Smagorinsky model should be avoided because it leads to non constant particle concentration.

Finally, tests with inertial particles have completed the picture. Initial poor results have been considerably improved by adding the modified SGS kinetic energy in the diffusion term, $\tilde{k}_{SGS}$, and changing the way we computed dissipation. Results are pretty good, especially with small inertial particles, which is in agreement with the good consistency of the limit case of fluid particles. Thus, we may only limit us to change and improve the part that takes into account the inertia of particles, keeping the rest of the model as a good starting model. We recall also that in the model formulated there are some constants that have been left unchanged in our simulations and that can be tuned with further tests to improve the results. Further tests with different grids are also needed to see if we have a better behaviour on finer grids.

The stochastic model studied in this work is probably the simplest that can be formulated because our aim is to refine the results without complicating too much the procedure. However this is not the only possible one, so it is not excluded that better results are possible with this class of models.

Finally we underline that this is a stochastic model based on one-point, one-time pdfs. To recover better coherent structures in the near wall layer it would be more powerful to have two-points pdfs in order to take into account spatial correlations. However this class of model is much more complicated. Another way to proceed is to use three different types of models depending on the position of the particle in the channel: if the particle is in the central region of the channel we can use the model adopted in this work, while if the particle is in the near wall region we have to chose between different possibilities. Having the particle the same probability to be within a sweep or an ejection, we simply assign to the particle a unitary wall normal velocity directed towards the center of the channel or towards the wall, each one with one half probability. This reasoning seems to work, but results for a 3D channel are not yet available.

Appendices

Appendix A

Diffusion processes

A diffusion process is a particular kind of stochastic process. It is a continuous-time Markov process with continuous sample paths.

Markov processes

Let $U(t)$ for $t \geq t_0$ be a stochastic process with one-time PDF $f(V;t)$. We introduce N times and consider the PDF of $U(t_N)$ conditioned on $U(t)$ at the earlier times, which is denoted by

$$f_{N-1}(V_N;t_N|V_{N-1},t_{N-1},V_{N-2},t_{N-2},...,V_1,t_1) \tag{A.0.1}$$

The PDF of U(t) conditioned on a single past time is denoted by, for example

$$f_1(V_N;t_N|V_{N-1},t_{N-1}) \tag{A.0.2}$$

By definition, if $U(t)$ is a Markov process then these conditional PDFs are equal

$$f_{N-1}(V_N;t_N|V_{N-1},t_{N-1},V_{N-2},t_{N-2},...,V_1,t_1) = f_1(V_N;t_N|V_{N-1},t_{N-1}) \tag{A.0.3}$$

This means that, given $U(t_{N-1}) = V_{N-1}$, knowledge of previous values provides no further information about the future value $U(t_N)$.

The Chapman-Kolmogorov equation

For any process, from the definition of conditional PDFs, we have

$$f_1(V_3;t_3|V_1,t_1) = \int_{-\infty}^{+\infty} f_2(V_3;t_3|V_2,t_2,V_1,t_1) f_1(V_2;t_2|V_1,t_1) dV_2 \tag{A.0.4}$$

For a Markov process, Eq. A.0.3 can be used to replace f_2 by $f_1(V_3;t_3|V_2,t_2)$ which leads to the *Chapman-Kolmogorov equation*

$$f_1(V_3;t_3|V_1,t_1) = \int_{-\infty}^{+\infty} f_1(V_3;t_3|V_2,t_2) f_1(V_2;t_2|V_1,t_1) dV_2 \tag{A.0.5}$$

Increments

A useful concept is the *increment* in a process: the increment in a positive time interval h is defined by

$$\Delta_h U(t) = U(t+h) - U(t) \tag{A.0.6}$$

It is important to note that h is positive and that the increment is defined forward in time. A process can be considered as a sum of its increments

$$U(t_N) = U(t_0) + \Delta_{t_1-t_0} U(t_0) + \Delta_{t_2-t_1} U(t_1) + ... + \Delta_{t_N-t_{N-1}} U(t_{N-1}) \tag{A.0.7}$$

The PDF of the increment $\Delta_h U(t)$, conditional on $U(t) = V$, is denoted by $g(\hat{V};h,V,t)$. If h is taken to be $t_3 - t_2$, then $U(t_2)$ can be re-expressed as

$$U(t_2) = U(t_3) - \Delta_h U(t_2) \tag{A.0.8}$$

and the first conditional PDF on the right-hand side of Eq. A.0.5 is

$$f_1(V_3;t_2+h|V_3-\hat{V},t_2) = g(\hat{V};h,V_3-\hat{V},t_2) \tag{A.0.9}$$

Thus the Chapman-Kolmogorov equation can be rewritten as

$$f_1(V;t_2+h|V_1,t_1) = \int_{-\infty}^{+\infty} g(\hat{V};h,V-\hat{V},t_2) f_1(V-\hat{V};t_2|V_1,t_1) d\hat{V} \tag{A.0.10}$$

Diffusion processes

There are qualitatively different kinds of continuous-time Markov processes, which are distinguished from each other by the behaviours of their increments $\Delta_h U(t)$ in the limit as h tends to zero. One defining property of a diffusion process is that its sample paths are continuous. More precisely, for every $\epsilon > 0$

$$\lim_{h\to 0} \frac{1}{h} P\{|\Delta_h U(t)| > \epsilon | U(t) = V\} = 0 \tag{A.0.11}$$

If they exist, the infinitesimal parameter of a process are defined by

$$B_n(V,t) = \lim_{h\to 0} \frac{1}{h} \langle [\Delta_h U(t)]^n | U(t) = V \rangle = \lim_{h\to 0} \frac{1}{h} \int_{-\infty}^{+\infty} \hat{V}^n g(\hat{V}; h, V, t) d\hat{V} \tag{A.0.12}$$

for $n = 1, 2, ...$ In addition to Eq. A.0.11, the defining properties of a diffusion process are that the *drift coefficient*

$$a(V,t) = B_1(V,t) \tag{A.0.13}$$

and the *diffusion coefficient*

$$b(V,t)^2 = B_2(V,t) \tag{A.0.14}$$

exist, and that the remaining infinitesimal parameters are zero

$$B_n(V,t) = 0, \quad for \;\; n \geq 3 \tag{A.0.15}$$

A diffusion process with $b(V,t) \neq 0$ is clearly nowhere differentiable, for the fact that $\langle [\Delta_h U(t)]^2 / h \rangle$ tends to a positive limit.

The Kramers-Moyal equation

In the Chapman-Kolmogorov equation, both g and f_1 on the right-hand side involve the argument $V - \hat{V}$. Expanding these quantities in a Taylor series about V yields

$$f_1(V; t_2 + h | V_1, t_1) = f_1(V; t_2 | V_1, t_1) + \int_{-\infty}^{+\infty} \sum_{n=1}^{\infty} \frac{(-\hat{V})^n}{n!} \frac{\partial^n}{\partial V^n} \Big[g(\hat{V}; h, V, t_2) f_1(V; t_2 | V_1, t_1) \Big] d\hat{V} \tag{A.0.16}$$

By dividing by h, taking the limit $h \to 0$, and using Eq. A.0.12, we obtain the *Kramers-Moyal equation*

$$\frac{\partial}{\partial t} f_1(V;t|V_1,t_1) = \sum_{n=1}^{\infty} \frac{(-1)^n}{n!} \frac{\partial^n}{\partial V^n} [B_n(V,t) f_1(V;t|V_1,t_1)] \tag{A.0.17}$$

The Fokker-Planck equation

For a diffusion process, all of the parameters B_n are zero, except for the drift and the diffusion. In this case the Kramers-Moyal equation reduces to the *Fokker-Planck* equation

$$\frac{\partial}{\partial t} f_1(V;t|V_1,t_1) = -\frac{\partial}{\partial V}[a(V,t) f_1(V;t|V_1,t_1)] + \frac{1}{2}\frac{\partial^2}{\partial V^2}[b(V,t)^2 f_1(V;t|V_1,t_1)] \tag{A.0.18}$$

The corresponding equation for the marginal PDF $f(V,t)$ is obtained by multiplying by $f(V_1,t_1)$ and integrating over V_1

$$\frac{\partial}{\partial t} f(V;t) = -\frac{\partial}{\partial V}[a(V,t) f(V;t)] + \frac{1}{2}\frac{\partial^2}{\partial V^2}[b(V,t)^2 f(V;t)] \tag{A.0.19}$$

Stochastic differential equations

Because diffusion processes are not differentiable, the standard tools of differential calculus cannot be applied. Instead of differential calculus, the appropriate method is the Ito calculus; and instead of being described by differential equations, diffusion processes are described by stochastic differential equations.

The *infinitesimal increment* of the process $U(t)$ is defined by

$$dU(t) = U(t+dt) - U(t) \tag{A.0.20}$$

where dt is a positive infinitesimal time interval. For the Wiener process in particular, we have the following properties:

- the process has independent increments: $(W(t_3) - W(t_2))$ and $(W(t_1) - W(t_0))$ are independent when $t_0 < t_1 < t_2 < t_3$
- the trajectories of the process are continuous functions (almost everywhere) but not differentiable
- the increments of the Wiener process $dW(t)$ are Gaussian random variables, with zero mean and a variance equal to dt.

$$\langle dW(t)\rangle = 0 \quad \langle (dW(t))^2\rangle = dt \tag{A.0.21}$$

Now consider the process $U(t)$ defined by the initial condition $U(t_0) = U_0$, and by the increment

$$dU(t) = a[U(t), t]dt + b[U(t), t]dW(t) \tag{A.0.22}$$

It is readily verified that the process $U(t)$ defined by this stochastic differential equation is a diffusion process, and as implied by the notation, the drift and diffusion coefficient are $a(V, t)$ and $b(V, t)^2$.

Appendix B

Subroutines

```
      SUBROUTINE PARTTRACK(DELTAT)

C#####################################################
      include '../FLW/precision.h'
      include '../FLW/param.h'
      include 'particleparam.h'
      include 'common.h'

C#####################################################
C These common blocks directly from FLOWSB are for the Fluid velocity
C field:
      COMMON / CONTRL / NTIMAX, CPMAX, XMON, YMON, ZMON, NSTART, NDUMP,
     :    NTIM, TIME, IT1, IT2, NPRINT(10), NPLOT(10), NLPLOT(10), LDUMP
      COMMON /ALFTOLD/ NTMOLD
      COMMON /BUFFER/ A(38*NXH*NY*NZ), LBUFF, LFIELD
      COMMON / POINT  / IPU, IPV, IPW, IPPR, IPS1(2), IPS2(2), IPS3(2),
     :    IPWO(17), IPST(2), IPT
      COMMON /TABLES/ AK1(NX), AK2(NY), DT, BETA,GAMA, RE, REU,
     &      GRADPX, GRADPY, AL1, AL2, WAIT(NZ), ISUB, UREF, CU1,CU2,
     &      RELAX, ZCOORD(NZ), SPARE(10), EDV, EDVOLD, VD, NORMAL,
     &      CT1, CT2, EDD, EDDOLD, XCOORD(NX)
C#####################################################
C Common block added for the Stochastic tracking
      COMMON / STOCHASTIC / U_OLD , DUDX_OLD , DVDX_OLD , DWDX_OLD ,
     &      DPDX_OLD
C#####################################################
C Common block added to see when stationary state is reached for the
C  particles
      COMMON / STATIONARY / SONDA
C#####################################################
C Fluid Velocity and Pressure field
      real*8, dimension(-1:NX+3,-1:NY+3,-1:NZ+2,3) :: U,DUDX,DVDX
      real*8, dimension(-1:NX+3,-1:NY+3,-1:NZ+2,3) :: DWDX,DPDX
      real*8, dimension(-1:NX+3,-1:NY+3,-1:NZ+2,3) :: U_OLD,DUDX_OLD
```

```
      real*8, dimension(-1:NX+3,-1:NY+3,-1:NZ+2,3) :: DVDX_OLD,DWDX_OLD
      real*8, dimension(-1:NX+3,-1:NY+3,-1:NZ+2,3) :: DPDX_OLD

      real*8 SONDA

      real*8 DELTAT

      real*8, dimension(npsetmax,1:npmax,3) :: TLAG , BX , PIIL
      real*8, dimension(npsetmax,1:npmax,3) :: TSUF , TSUP
      real*8, dimension(npsetmax,1:npmax) :: TAUP
      integer :: NOR , STATSPACE

*****************************************************************
C Assigning the time step from Excute1
      DTP = DELTAT

C TIMEP is actual instant in non-dimensional (+) units
C                (TIME  is in non-dimensional (-) units
C                [taken from FLOWSB] )
      TIMEP = TIME*RE

      CALL GETFLUIDVEL(U)
      CALL CALCULATEDERIVATIVES(DUDX,DVDX,DWDX,DPDX)

      DO ID = 1,9
        CALL NORMALEN(nm, VAGAUS(1,ID))
      ENDDO

      NOR=0

      IF (INTERPART.EQ.0) THEN

      CALL LAGCAR2(U_OLD,DUDX_OLD,DVDX_OLD,DWDX_OLD,DPDX_OLD,
     &                TLAG,BX,PIIL,1)

      ELSEIF (INTERPART.EQ.1) THEN

      CALL LAGCAR2NGP(U_OLD,DUDX_OLD,DVDX_OLD,DWDX_OLD,DPDX_OLD,
     &                TLAG,BX,PIIL,1)

      ELSEIF (INTERPART.EQ.2) THEN

      CALL LAGCAR2INT(U_OLD,DUDX_OLD,DVDX_OLD,DWDX_OLD,DPDX_OLD,
     &                TLAG,BX,PIIL,1)

      ENDIF

 10   CONTINUE

      NOR=NOR+1

      IF (NORDRE.EQ.1) THEN
        CALL LAGES(U_OLD,DUDX_OLD,DVDX_OLD,DWDX_OLD,
```

```
     &                    TLAG,BX,PIIL,TAUP)
          CALL REBOUNDS(U_OLD)
        ELSE
          CALL LAGES2(U,DUDX,DVDX,DWDX,DPDX,U_OLD,
     &                DUDX_OLD,DVDX_OLD,DWDX_OLD,
     &                TLAG,BX,PIIL,TSUP,TSUF,TAUP,NOR)

          IF (NOR.EQ.1) THEN
            CALL REBOUNDS(U_OLD)
          ELSE
            CALL REBOUNDS(U)
          ENDIF

        ENDIF

        IF ((NOR.EQ.1).AND.(NORDRE.EQ.2)) GOTO 10

        DO ID=1,3
        DO I=1,NX
        DO J=1,NY
        DO K=1,NZ
          U_OLD(I,J,K,ID)=U(I,J,K,ID)
          DUDX_OLD(I,J,K,ID)=DUDX(I,J,K,ID)
          DVDX_OLD(I,J,K,ID)=DVDX(I,J,K,ID)
          DWDX_OLD(I,J,K,ID)=DWDX(I,J,K,ID)
          DPDX_OLD(I,J,K,ID)=DPDX(I,J,K,ID)
        ENDDO
        ENDDO
        ENDDO
        ENDDO

C**********************************************************************
C---WRITE SONDA FILE-------------------------------------------------------C
        IF (NSTEP.EQ.0) THEN
          OPEN(199,FILE='sonda_stat',STATUS='UNKNOWN')
        ENDIF
        IF((MOD((NTIM-NTMOLD),10).EQ.0).OR.(NTIM.EQ.NTIMAX))THEN
           WRITE(199,*) NTIM,SONDA
        ENDIF
        IF (NTIM.EQ.NTIMAX) THEN
          CLOSE(199,STATUS='KEEP')
        ENDIF

C**********************************************************************
C---WRITE OUTPUT FILES-----------------------------------------------------C
        IF(OUT.GT.0.)THEN
            IF((MOD((NTIM-NTMOLD),OUT).EQ.0).OR.(NTIM.EQ.NTIMAX))THEN
             WRITE(*,*) 'TRY: DUMPING PARTICLE-OUTPUTFILE',NTIM
             CALL PARTOUT
            ENDIF
        ENDIF

C**********************************************************************
```

```
C---WRITE OUTPUT FILES-------------------------------------------------------C
      WRITE(*,*)ICONC
      IF(ICONC.GT.0.)THEN
            WRITE(*,*) 'TRY: Calculating concentration profiles',NTIM
            CALL PARTICLE_CONCENTRATION(NTIM,NTMOLD,NTIMAX,TIME)
      ENDIF

**********************************************************************
C---WRITE RESTART FILES------------------------------------------------------C
      IF(REST.GT.0.)THEN
           IF((MOD((NTIM-NTMOLD),REST).EQ.0).OR.(NTIM.EQ.NTIMAX))THEN
            WRITE(*,*) 'TRY: DUMPING PARTICLE-RESTARTFILE',NTIM
            CALL PARTWRITEREST(NTIM)
           ENDIF
      ENDIF

**********************************************************************
C---CALCULATE STATISTICS-----------------------------------------------------C
      STATSPACE = 10
      IF((MOD((NTIM-NTMOLD),STATSPACE).EQ.0).OR.(NTIM.EQ.NTIMAX).OR.
     &    (NSTEP.EQ.0))THEN
          CALL PARTSTATISTICS(STATSPACE)
      ENDIF

**********************************************************************
C---UPDATE OF PARTICLE POSITION AND VELOCITY---------------------------------C
      DO npset=1,nsetnumber
       DO n=1,nm
        DO j=1,3
         P0(j,npset,n) = P(j,npset,n)
         V0(j,npset,n) = V(j,npset,n)
        ENDDO
       ENDDO
      ENDDO


      RETURN
      END


      SUBROUTINE LAGES(U,DUDX,DVDX,DWDX,TLAG,BX,PIIL,TAUP)
C######################################################################

      include '../FLW/precision.h'
      include '../FLW/param.h'
      include 'particleparam.h'
      include 'common.h'

C######################################################################
C Fluid Velocity field
      real*8, dimension(-1:NX+3,-1:NY+3,-1:NZ+2,3) :: U,DUDX,DVDX,DWDX

      real*8, dimension(npsetmax,1:npmax,3) :: TLAG , BX , PIIL
      real*8 :: VITF
```

```
      real*8 :: AA , BB , CC , DD , EE
      real*8 :: AUX1 , AUX2 ,AUX3 , AUX4 , AUX5 , AUX6
      real*8 :: AUX7 , AUX8 , AUX9 , AUX10 , AUX11
      real*8 :: TER1F , TER2F , TER3F
      real*8 :: TER1P , TER2P , TER3P , TER5P
      real*8 :: TER1X , TER2X , TER3X , TER5X
      real*8 :: TCI
      real*8 :: GAMA2 , OMEGAM , OMEGA2
      real*8 :: GRGA2 , GAGAM , GAOME
      real*8 :: P11 , P21 , P22 , P31 , P32 , P33
      real*8 :: REYP
      real*8 :: TAUP(npsetmax,npmax)
      integer :: ID , NPSET , N
      integer :: IPARTX , IPARTY , IPARTZ

**********************************************************************
      DO ID = 1,3

      DO NPSET=1,nsetnumber
        DO N = 1,nm
          ETTPA(ID,NPSET,N)=ETTP(ID,NPSET,N)
        ENDDO
      ENDDO

      ENDDO

      DO ID = 1,3

      DO NPSET=1,nsetnumber
        DO N = 1,nm

          CALL LOCATEPART(IPARTX,IPARTY,IPARTZ,
     &                    P0(1,NPSET,N),P0(2,NPSET,N),P0(3,NPSET,N))

            IF (INTERPART.EQ.0) THEN

            IF (ID.EQ.1) VITF = U(IPARTX,IPARTY,IPARTZ,ID)
            IF (ID.EQ.2) VITF = U(IPARTX,IPARTY,IPARTZ,ID)
            IF (ID.EQ.3) VITF = U(IPARTX,IPARTY,IPARTZ,ID)

            ELSEIF (INTERPART.EQ.1) THEN

            IF (ID.EQ.1) VITF = ( U(IPARTX,IPARTY,IPARTZ,ID) +
     &                            U(IPARTX+1,IPARTY,IPARTZ,ID) +
     &                            U(IPARTX,IPARTY+1,IPARTZ,ID) +
     &                            U(IPARTX+1,IPARTY+1,IPARTZ,ID) +
     &                            U(IPARTX,IPARTY,IPARTZ+1,ID) +
     &                            U(IPARTX+1,IPARTY,IPARTZ+1,ID) +
     &                            U(IPARTX,IPARTY+1,IPARTZ+1,ID) +
     &                            U(IPARTX+1,IPARTY+1,IPARTZ+1,ID) ) / 8.D0
            IF (ID.EQ.2) VITF = ( U(IPARTX,IPARTY,IPARTZ,ID) +
     &                            U(IPARTX+1,IPARTY,IPARTZ,ID) +
     &                            U(IPARTX,IPARTY+1,IPARTZ,ID) +
```

```
     &                          U(IPARTX+1,IPARTY+1,IPARTZ,ID) +
     &                          U(IPARTX,IPARTY,IPARTZ+1,ID) +
     &                          U(IPARTX+1,IPARTY,IPARTZ+1,ID) +
     &                          U(IPARTX,IPARTY+1,IPARTZ+1,ID) +
     &                          U(IPARTX+1,IPARTY+1,IPARTZ+1,ID) ) / 8.D0
           IF (ID.EQ.3) VITF = ( U(IPARTX,IPARTY,IPARTZ,ID) +
     &                          U(IPARTX+1,IPARTY,IPARTZ,ID) +
     &                          U(IPARTX,IPARTY+1,IPARTZ,ID) +
     &                          U(IPARTX+1,IPARTY+1,IPARTZ,ID) +
     &                          U(IPARTX,IPARTY,IPARTZ+1,ID) +
     &                          U(IPARTX+1,IPARTY,IPARTZ+1,ID) +
     &                          U(IPARTX,IPARTY+1,IPARTZ+1,ID) +
     &                          U(IPARTX+1,IPARTY+1,IPARTZ+1,ID) ) / 8.D0

           ELSEIF (INTERPART.EQ.2) THEN

           IF (ID.EQ.1) VITF = UPART(1,NPSET,N)
           IF (ID.EQ.2) VITF = UPART(2,NPSET,N)
           IF (ID.EQ.3) VITF = UPART(3,NPSET,N)

           ENDIF

C#######################################################################
C Calculation of TiCi

           TCI = PIIL(NPSET,N,ID) * TLAG(NPSET,N,ID) + VITF

C#######################################################################
C Calcolo dei coefficienti/termini deterministici

           REYP = ( ( ETTPA(1,NPSET,N) - V0(1,NPSET,N) )**2. +
     &              ( ETTPA(2,NPSET,N) - V0(2,NPSET,N) )**2. +
     &              ( ETTPA(3,NPSET,N) - V0(3,NPSET,N) )**2. )**(0.5)
     &            * DPP(NPSET)

           TAUP(NPSET,N) = TAUPP(NPSET) / (1.D0 + 0.15D0*(REYP**0.687D0))

           AUX1 = EXP( -DTP / TAUP(NPSET,N))
           AUX2 = EXP( -DTP / TLAG(NPSET,N,ID))
           AUX3 = TLAG(NPSET,N,ID) / (TLAG(NPSET,N,ID)-TAUP(NPSET,N))

           AUX4 = TLAG(NPSET,N,ID) / (TLAG(NPSET,N,ID)+TAUP(NPSET,N))
           AUX5 = TLAG(NPSET,N,ID) * (1.D0-AUX2)
           AUX6 = BX(NPSET,N,ID) * BX(NPSET,N,ID) * TLAG(NPSET,N,ID)

           AUX7 = TLAG(NPSET,N,ID) - TAUP(NPSET,N)
           AUX8 = BX(NPSET,N,ID) * BX(NPSET,N,ID) * AUX3**2

C Terms for the trajectories

           AA = TAUP(NPSET,N) * (1.D0 - AUX1)
           BB = (AUX5 - AA) * AUX3
           CC = DTP - AA - BB
```

```
C ETTPA is the fluid modelled velocity
              TER1X = AA * V0(ID,NPSET,N)
              TER2X = BB * ETTPA(ID,NPSET,N)
              TER3X = CC * TCI

C Terms for the fluid

              TER1F = ETTPA(ID,NPSET,N) * AUX2
              TER2F = TCI * (1.D0-AUX2)

C Terms for particle velocity

              DD = AUX3 * (AUX2 - AUX1)
              EE = 1.D0 - AUX1

              TER1P = V0(ID,NPSET,N) * AUX1
              TER2P = ETTPA(ID,NPSET,N) * DD
              TER3P = TCI * (EE-DD)

C Calculation of the correlation matrix

              GAMA2  = 0.5D0 * (1.D0 - AUX2*AUX2 )
              OMEGAM = 0.5D0 * AUX4 * ( AUX5 - AUX2*AA )
     &                 -0.5D0 * AUX2 * BB
              OMEGAM = OMEGAM * SQRT(AUX6)

              OMEGA2 = AUX7
     &                 * (AUX7*DTP - 2.D0 *
     &                   (TLAG(NPSET,N,ID)*AUX5-TAUP(NPSET,N)*AA))
     &               + 0.5D0 * TLAG(NPSET,N,ID) * TLAG(NPSET,N,ID) * AUX5
     &                 * (1.D0 + AUX2)
     &               + 0.5D0 * TAUP(NPSET,N) * TAUP(NPSET,N) * AA *
     &                 (1.D0+AUX1) - 2.D0 * AUX4 * TLAG(NPSET,N,ID) *
     &                  TAUP(NPSET,N) * TAUP(NPSET,N) * (1.D0 - AUX1*AUX2)

              OMEGA2 = AUX8 * OMEGA2

              IF (ABS(GAMA2).GT.0) THEN

                P21 = OMEGAM / SQRT(GAMA2)
                P22 = OMEGA2 - P21**2

                P22 = SQRT( MAX(0.D0,P22) )

              ELSE
                P21 = 0.D0
                P22 = 0.D0
              ENDIF

              TER5X = P21 * VAGAUS(N,ID) + P22 * VAGAUS(N,3+ID)
```

```
            P11 = SQRT( GAMA2*AUX6 )
            TER3F = P11*VAGAUS(N,ID)


            AUX9  = 0.5D0 * TLAG(NPSET,N,ID) * (1.D0 - AUX2*AUX2)
            AUX10 = 0.5D0 * TAUP(NPSET,N) * (1.D0 - AUX1*AUX1)
            AUX11 = TAUP(NPSET,N) * TLAG(NPSET,N,ID) *
     &              (1.D0 - AUX1*AUX2)
     &               / (TAUP(NPSET,N) + TLAG(NPSET,N,ID))

            GRGA2 = (AUX9 - 2.D0*AUX11 + AUX10) * AUX8
            GAGAM = (AUX9 - AUX11) * (AUX8 / AUX3)
            GAOME = ( (TLAG(NPSET,N,ID) - TAUP(NPSET,N)) * (AUX5 - AA)
     &             - TLAG(NPSET,N,ID) * AUX9 - TAUP(NPSET,N) * AUX10
     &             + (TLAG(NPSET,N,ID) + TAUP(NPSET,N)) * AUX11) * AUX8

            IF(P11.GT.0) THEN
              P31 = GAGAM / P11
            ELSE
              P31 = 0.D0
            ENDIF

            IF(P22.GT.0) THEN
              P32 = (GAOME-P31*P21) / P22
            ELSE
              P32 = 0.D0
            ENDIF

            P33 = GRGA2 - P31**2 - P32**2

            P33 = SQRT( MAX(0.D0,P33) )

            TER5P = P31 * VAGAUS(N,ID)
     &            + P32 * VAGAUS(N,3+ID)
     &            + P33 * VAGAUS(N,6+ID)


C Time advancement
C---> trajectory
            P(ID,NPSET,N) = P0(ID,NPSET,N)
     &                      + TER1X + TER2X + TER3X + TER5X

C---> fluid velocity

            ETTP(ID,NPSET,N) = TER1F + TER2F + TER3F

C---> particle velocity

            V(ID,NPSET,N) = TER1P + TER2P + TER3P + TER5P

          ENDDO
        ENDDO
```

```
ENDDO


RETURN
END
```

Bibliography

[1] C. Marchioli, M.V.Salvetti and A. Soldati "Some issues concerning Large-Eddy Simulation of inertial particle dispersion in turbulent bounded flows", *Physics of Fluids*, 20:040603, 2002.

[2] J.G.M. Kuerten, A.W. Vreman "Can turbophoresis be predicted by large-eddy simulation?", *Physics of Fluids*, 17:011701, 2005.

[3] C. Marchioli, M.V.Salvetti and A. Soldati "Appraisal of energy recovering sub-grid scale models for large-eddy simulation of turbulent dispersed flows", *Acta Mechanica*, 201:277-296, 2008.

[4] F. Bianco, S. Chibbaro, C. Marchioli, M.V.Salvetti and A. Soldati "Intrinsic filtering errors of Lagrangian particle tracking in LES flow fields", *Physics of Fluids*, 24:045103, 2012.

[5] M.R. Maxey and J.J. Riley "Equation of motion for a small rigid sphere in a nonuniform flow", *Physics of Fluids*, 26:883, 1983.

[6] C. Marchioli, A. Soldati "Mechanisms for particle transfer and segregation in a turbulent boundary layer", *Journal of fluid Mechanics*, 468:283–315, 2002.

[7] C. Marchioli, A. Giusti, M.V. Salvetti, A. Soldati "Direct numerical simulation of particle wall transfer and deposition in upward turbulent pipe flow", *International journal of Multiphase flow*, 29(6):1017–1038, 2003.

[8] J.W. Cleaver, B. Yates "A sub layer model for the deposition of particles from a turbulent flow", *Chemical Engineering Science*, 30(8):983–992, 1975.

[9] C. Marchioli "Mechanisms for transfer, segregation and deposition of heavy particles in turbulent boundary layers", *PhD Thesis*, 2003.

[10] W. Shoppa, F. Hussain "Genesis and dynamics of coherent structures in near-wall turbulence", *Advances in Fluid Mechanics*, 15:385-422, 1997.

[11] A. Soldati, C. Marchioli "Prospects for modulation of turbulent boundary layer by EHD flows", *CISM Courses and Lectures*, 415:119-160, 2001.

[12] S.B. Pope "Turbulent flows", *Cambridge University press*, 2000.

[13] P.J. Colucci, F.A. Jaberi, P. Givi, S.B. Pope "Filtered density function for large eddy simulation of turbulent reacting flows" *Physics of Fluids*, 10,499, 1998

[14] L.Y.M. Gicquel, P. Givi, F.A. Jaberi, S.B. Pope "Velocity filtered density function for large eddy simulation of turbulent flows", *Physics of Fluids*, 14(3), 2002

[15] M.R.H. Sheikhi, T.G. Drozda, P. Givi, S.B. Pope "Velocity-scalar filtered density function for large eddy simulation of turbulent flows", *Physics of Fluids*, 15,2321, 2003

[16] J.P. Minier, E. Peirano "The pdf approach to polydispersed turbulent two-phase flows", *Physics Reports*, 352(1-3):1-214, 2001.

[17] P. Moin, J. Kim "Numerical investigation of turbulent channel flow", *Journal of Fluid Mechanics*, 118:341-377, 1982.

[18] E. Peirano, S. Chibbaro, J. Pozorski, J.P. Minier "Mean-field/PDF numerical approach for polydispersed tuebulent two-phase flows", *Progress in Energy and Combustion Science*, 32:315-371, 2006.

[19] A.S. Berrouk, D. Laurence, J.J. Riley, D.E. Stock "Stochastic modelling of inertial particle dispersion by subgrid motion for LES of high Reynolds number pipe flow", *Journal of Turbulence*, 8(50), 2007.

[20] J.P. Minier, E. Peirano, S. Chibbaro "Weak first- and second-order numerical schemes for stochastic differential equations appearing in Lagrangian two-phase flow modeling", *Monte Carlo Methods and App.*, 9(2):93-133, 2003.

[21] D.C. Haworth, S.B. Pope "A second-order Monte-Carlo method for the solution of the Ito stochastic differential equation", *Stochastic Analysis and Applications*, 4(2):151-186, 1986.