EDF R&D

Fluid Dynamics, Power Generation and Environment Department
Single Phase Thermal-Hydraulics Group

6, quai Watier
F-78401 Chatou Cedex

Tel: 33 1 30 87 75 40
Fax: 33 1 30 87 79 16

SEPTEMBER 2017

Code_Saturne documentation

Code_Saturne version 4.0 tutorial:
Particle Dispersion in a Turbulent Pipe Flow

contact: saturne-support@edf.fr

TABLE OF CONTENTS

Chapter I

Introduction

1 Tutorial Components

This tutorial makes use of:

- The SALOME [1] platform for geometry generation, meshing, and post-processing
- *Code_Saturne* [2, 3] for CFD calculations
- Reference [4] for comparison with published results

To work through this tutorial you will need a computer on which these two software applications are already available or on which you have permission to install them.

You will also need to know how to create and setup a *Code_Saturne* case with the CFDStudy module. For instructions on how to do so, please see [5].

2 Tutorial Structure

This tutorial focuses on the modelling of particle dispersion in turbulent pipe flow using the Lagrangian module of *Code_Saturne*.

This tutorial is made of five parts:

- Part 1 (Section II) presents the Arnason experimental set-up, the flow physics and the operating conditions.
- Part 2 (Section III) illustrates how to create the computational doamain (geometry and mesh) using the HexaBlock module in SALOME.
- Part 3 (Section IV) shows how to set-up the single phase flow case in *Code_Saturne* for the RANS simulation.
- Part 4 (Section V) describes how to set-up a Lagrangian simulation on a frozen velocity field.
- Part 5 (Section VI) presents the results of the numerical simulation compared with [4].

Chapter II

Part 1 - Particle Dispersion in a Turbulent Pipe Flow

1 Case Description

The experiment was carried out by Gylfi Arnason at Washington University [4], in order to assess the impact of flow turbulence on particle dispersion in dilute turbulent two phase flows. Laser Doppler anemometry was used for the first time in such an experiment.

The experimental set-up consists of a vertical pipe through which air is flowing at a constant flow rate. Glass beads are injected into the flow at a fixed distance downstream of the air inlet. The beads are then transported and diffused by the air in the pipe.

The test rig dimensions are described next.

1.1 Test Rig Dimensions

The pipe used in the experiment and the position of the injection point, where the origin of the reference frame is located, are shown in Figure II.1.

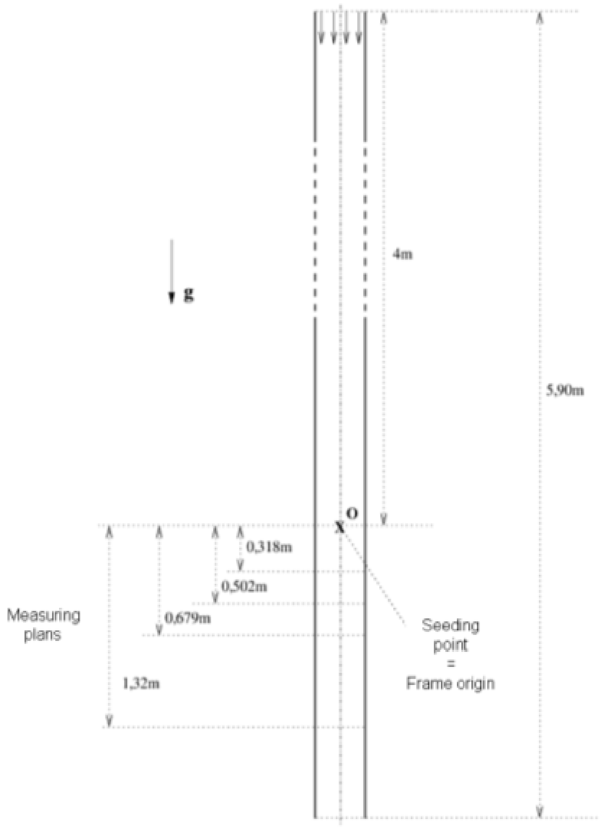


Figure II.1: Schematic description of the Arnason experimental set-up [4]

The dimensions of the pipe and the distance from the air inlet to the injection point are listed in Table II.1 for clarity.

Pipe Internal Diameter(m)	**Pipe Length(m)**	**Distance from Inlet to Injection (m)**
0.09	5.9	4.0

Table II.1: Arnason pipe dimensions.

As can be seen in Figure II.1, four measuring planes were used to obtain measurements of the air flow and the glass beads. These four planes are located downstream of the bead injection point with their positions relative to this point listed in Table II.2.

Plane	**Distance from injector(m)**
1	0.318
2	0.502
3	0.679
4	1.320

Table II.2: Distance of the measuring planes downstream of the injector.

The flow physics are described next.

1.2 Flow Physics

The flow in the pipe is incompressible, fully developed, turbulent, with the air carrier phase transporting and diffusing a second phase. Given that the second phase is dilute with respect to the carrier fluid, it is possible to assume that the glass beads are influenced by the flow of air but have no influence on the magnitude and flow direction of the carrier fluid. This simplified modelling is known as one-way coupling.

The test rig operating conditions are described next.

1.3 Operating Conditions

The experimental test rig was operating with the following conditions:

- The maximum air velocity is of the order of 9.56m/s
- The Reynolds number based on this maximum velocity is 50×10^3
- The Reynolds number based on the mean velocity is 42×10^3

The air temperature at inlet to the pipe is not provided in [4] so it is assumed to be $10°C$.

Two sets of experiments are carried out with two different sets of glass beads and identical carrier fluid conditions. The first set uses a bead diameter of $5\mu m$ and the second set a bead diameter of $57\mu m$.

The fluid properties are described next.

1.4 Fluid Properties

The fluid properties at the inlet temperature of $10°C$ are listed in Table IV.2.

$\rho(kg/m^3)$	$\mu(Pa.s)$
1.2361	$1.78 * 10^{-5}$

Table II.3: Fluid properties.

The properties of the glass beads are described next.

1.5 Glass Beads

The properties of the $5\mu m$ glass beads are listed in Table II.4.

Mean diameter	**Diameter Deviation**	**Density**
$d_p^\mu(\mu m)$	$\sigma_p(\mu m)$	$\rho(kg/m^3)$
5.0	1.0	2475

Table II.4: Glass bead properties.

The particle diameter d_p is calculated from the mean diameter and the deviation by $d_p = d_p^\mu + \varepsilon * \sigma_p$, where ε is a random variable which follows a normal law.

The boundary conditions are discussed next.

1.6 Boundary Conditions

Three carrier fluid boundary conditions are used in this study: inlet, outlet and wall. For the particles, the only boundary condition used is wall. Figure II.2 illustrates the location of these boundary conditions and the flow direction (blue arrows).

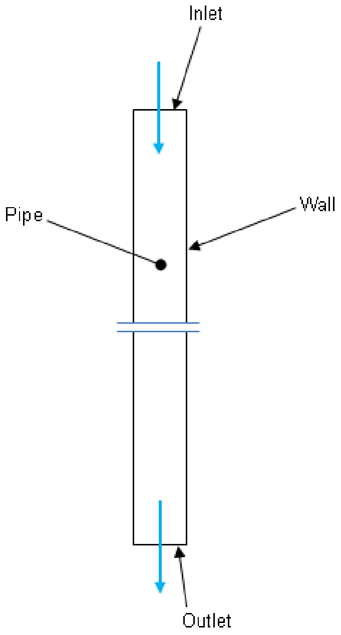


Figure II.2: Location of the boundary conditions.

Table II.5 lists the various boundary values applied for the carrier fluid (air) and the particles.

Phase	**Boundary Conditions and Values**		
	Inlet	**Outlet**	**Wall**
Air	$u = 0$ $v = 0$ $w = -V_{max}(1.0 - \frac{0.4 \times r^2}{0.002025})$ $D_h = 0.09m$	Standard	Wall
Particles	-	-	Rebound

Table II.5: Boundary conditions and values for the air and the particles.

The inlet boundary condition corresponds to a Reynolds number of 42000 based on the mean velocity which itself corresponds to a mass flow rate of 0.06291kg/s. A correction factor of 1.0488 is used to maintain the Reynolds number.

In the Arnason experiment, the particles are injected along the centre line of the pipe at the reference point. Since the particle injection velocity is not given in [4], in this tutorial we assume that it is equal to the local fluid velocity.

Also, the results presented in [4] do not stipulate the number of particles injected, the results only give statistical data. For the simulations in this tutorial, 1000 particles are injected per time step in order to have a sufficient number of particles to post process.

Lastly, in the CFD model the turbulence at the inlet boundary is calculated directly by *Code_Saturne* using the hydraulic diameter, D_h, specified.

2 One-Way Coupling CFD Modelling

The one-way coupling simulation of the two-phase flow of the air and the glass beads is broken into two steps. First, the flow of air alone is simulated. This will be used as the background flow on which the particles are injected. Then, in the second step, the flow of the particles on top of this air flow field is calculated.

3 Creating the *Code_Saturne* Study and Cases

A *Code_Saturne* study called *Arnason* and two cases are created. One for the first set of single-phase calculations, which we call *RANS_k_eps_LP*, and the other for the Lagrangian particulate simulations on the frozen flow field, which we call *RANS_k_eps_LP_5UM*.

The study and the cases are created using the procedure described in Part I of tutorial 1 [5]. Start SALOME with the command *code_saturne salome*, select the CFDStudy module, and go through all the steps detailed in [5] to:

- Create the CFD case structure with the CFDStudy module
- Save the new file as 'Arnason'
- Customise the background settings

In the end, you should end up with the directory structure (Figure II.3) shown in the Object Browser tab, displaying the study and the two cases.

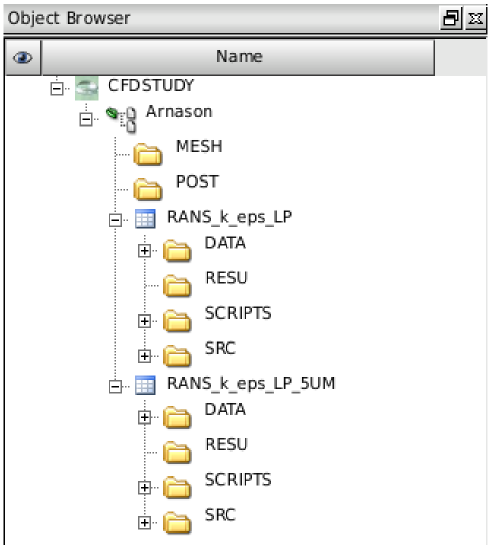


Figure II.3: *Arnason* Study, *RANS_k_eps_LP* and *RANS_k_eps_LP_5UM* case File Structure

The case is then ready to be set up.

Chapter III

Part 2 - Creating the Computational Domain with SALOME/HexaBlock

1 What you Will Learn

In this second part of this tutorial you will learn how to create the computational domain using the HexaBlock module in SALOME. This module significantly simplifies the generation of pipe type flow domains and ensures a conformal hexahedral mesh. The HexaBlock module will be used both to generate the pipe geometry and to create the volume mesh.

2 Creating the Geometry and Mesh

Once the files structure is created by following the step in 3, move to the HexaBlock module in SALOME. .

Open the HexaBlock module either by the scroll down-menu or the icon, represented as a green 3D square in wireframe, on the right side of Figure III.1.

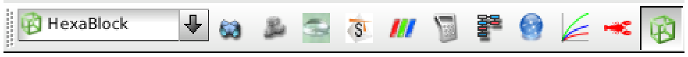


Figure III.1: Selecting the HexaBlock module.

Having selected the HexaBlock module, you will now generate the flow domain. You will create two cylinders which will subsequently define two zones in the flow domain, upstream and downstream of the injection plane, respectively. The reason for creating two cylinders is that the mesh upstream of the injection point will be coarser than that downstream of it. This will not have an impact on solution accuracy but will reduce the overall computational time.

First be sure that all the HexaBlock functionalities are displayed as shown in Figure III.2.

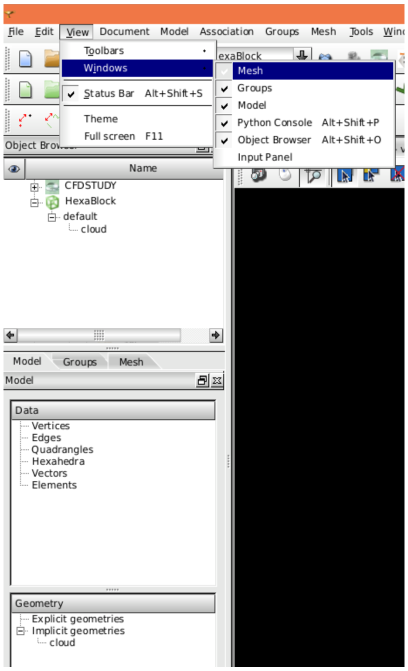


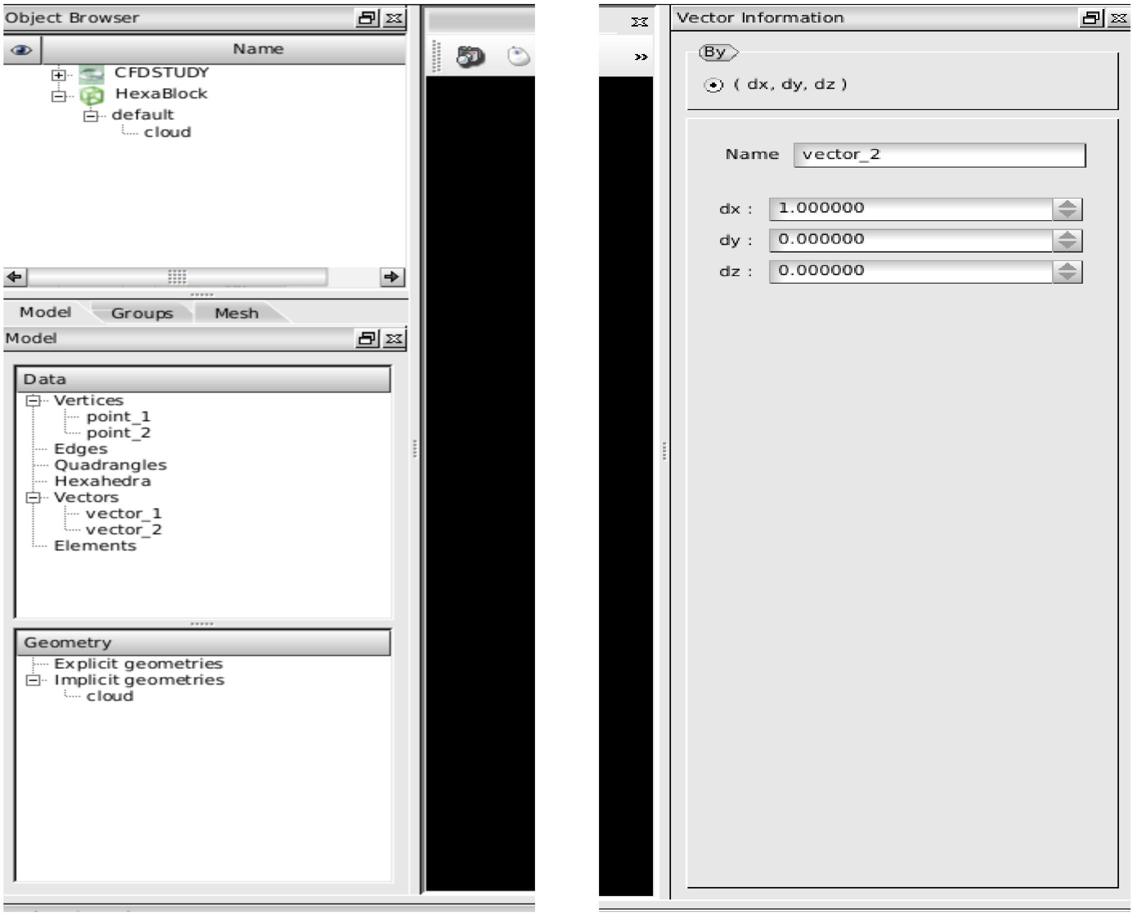
Figure III.2: Displaying Model, Groups and Mesh tabs.

It is very important to have the different tabs 'Model, Groups and Mesh' open in order to track any modification or object that you will add to the study.

It is advisable to save the 'MixingTee.hdf' file at regular intervals during this tutorial and certainly prior to and upon completion of any significant modifications. In doing so, your latest work is backed-up and, should mistakes happen that are irrecoverable, it is possible to exit SALOME and restart it from the last file save.

In the Model tab, create two points which will be the base of the two cylinders by selecting ***Model⇒ Construction⇒Add Vertex***. Name the two points 'point_1' and 'point_2'. The coordinates of the two points are (0.0; 0.0; 4.0) for 'point_1' and (0.0; 0.0; 0.0) for 'point_2'. The vertex 'point_1' is located at the centre of the inlet surface and the vertex 'point_2' is at the centre of the particle injection plane.

Now you will create the vectors which will be used with the points to create the two cylinders or zones. Create the vectors by selecting: ***Model⇒Construction⇒Add Vector***. For the first vector, change the name to 'vector_1' with the following orientation (0.0; 0.0; -1.0) then click 'Apply'. For the second vector, change the name to 'vector_2' with the following orientation (1.0; 0.0; 0.0) and click on 'Apply' (Figure III.3).

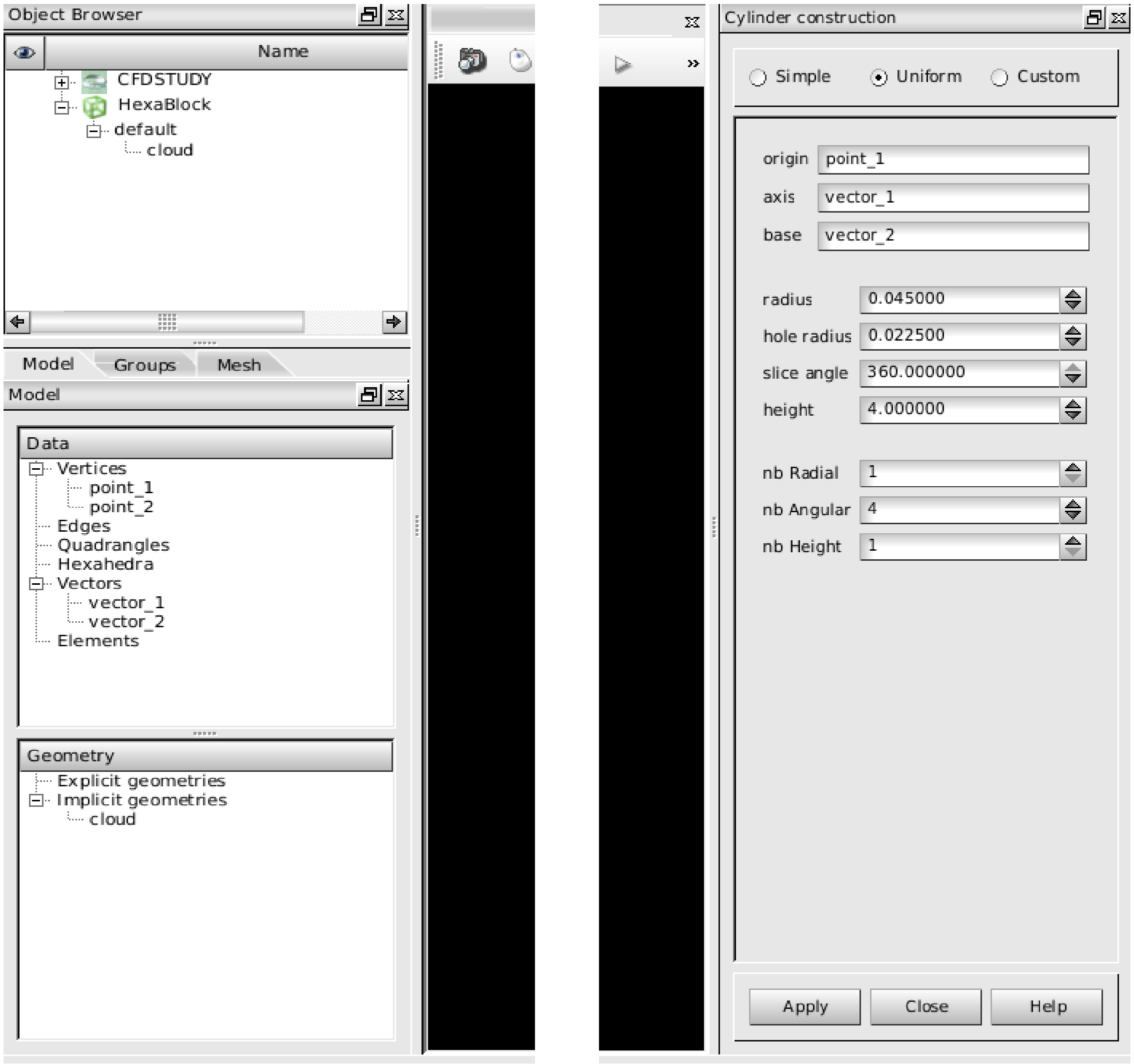


Name: vector_2
dx : 1.000000
dy : 0.000000
dz : 0.000000

Figure III.3: Creating the second vector.

Having created the vertices and vectors, you will now create the two cylinders which will represent the flow domain for the numerical model. For the first cylinder, open the cylinder tool by selecting: ***Model⇒Construction⇒Make Cylinder***.

For the first cylinder, select 'uniform' then fill the field 'origin' by 'point_1', axis by 'vector_1' and base by 'vector_2'. Enter a radius of 0.045m, a hole radius of 0.0225m and its height of 4m. Enter the value 1 for 'nb Radial' and 'nb Height' and the value 4 for 'nb Angular' as shown in Figure III.4. Then click 'Apply'.



○ Simple ● Uniform ○ Custom

origin point_1
axis vector_1
base vector_2

radius 0.045000
hole radius 0.022500
slice angle 360.000000
height 4.000000

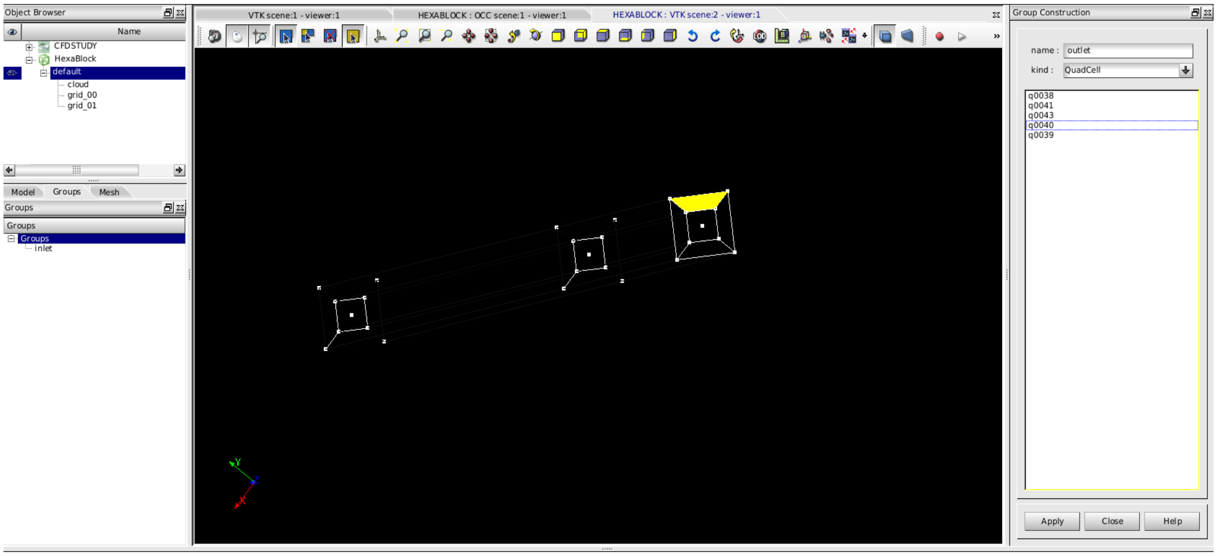
nb Radial 1
nb Angular 4
nb Height 1

Figure III.4: Creating the first cylinder.

For the second cylinder, replace 'point_1' by 'point_2' in the 'origin' field and modify the height of 1.9m. Then click 'Apply'. If the two cylinders have been created correctly, eight propagations should be visible in the Mesh tab. Before setting up different laws of propagation, move to the 'Groups' tab.

3 Creating Groups

Having generated the pipe flow domain, the next step is to create the groups that will be used as boundary conditions by the numerical model. Select ***Model⇒Add group***, then select 'QuadCell' and create the three boundary conditions (inlet, outlet, wall) by selecting the faces in the VTK viewer. To erase an unwanted face, select the face and press the key 'x'.



name : outlet
kind : QuadCell

Figure III.5: Creating groups.

4 Meshing

Having generated the groups for the T-Junction geometry, the next step is to generate the volume mesh. Move to the 'Mesh' window by clicking on the 'Mesh' tab.

At this point, the HexaBlock module will have automatically performed certain tasks for the mesh generation process. These tasks relate to the "propagations" that are generated by HexaBlock which are listed in Figure III.6 in the 'Mesh' window (Propagation0 to Propagation7). The term "propagation" relates to how the volume mesh, which is defined by a series of Laws, will be propagated throughout the flow domain.

Here, HexaBlock has applied propagations to the edges of the flow domain so that they can subsequently be used by the Laws that you will define as a mean of discretising an edge (or a group of edges) in order to define a number of cells that should be generated along these edges.

In order to verify which propagations are applied to which edges, the propagations generated by HexaBlock can be highlighted in the graphics window by clicking on them in the 'Mesh' window.

In the 'Mesh' tab, you will now define the laws (***Mesh⇒Add law***) which will then be associated with propagations for generating the volume meshes in the flow domain. For the pipe in this tutorial, you will need to define four laws to generate the volume mesh. The four laws used for the single phase and two-phase simulations are listed in the Table III.1.

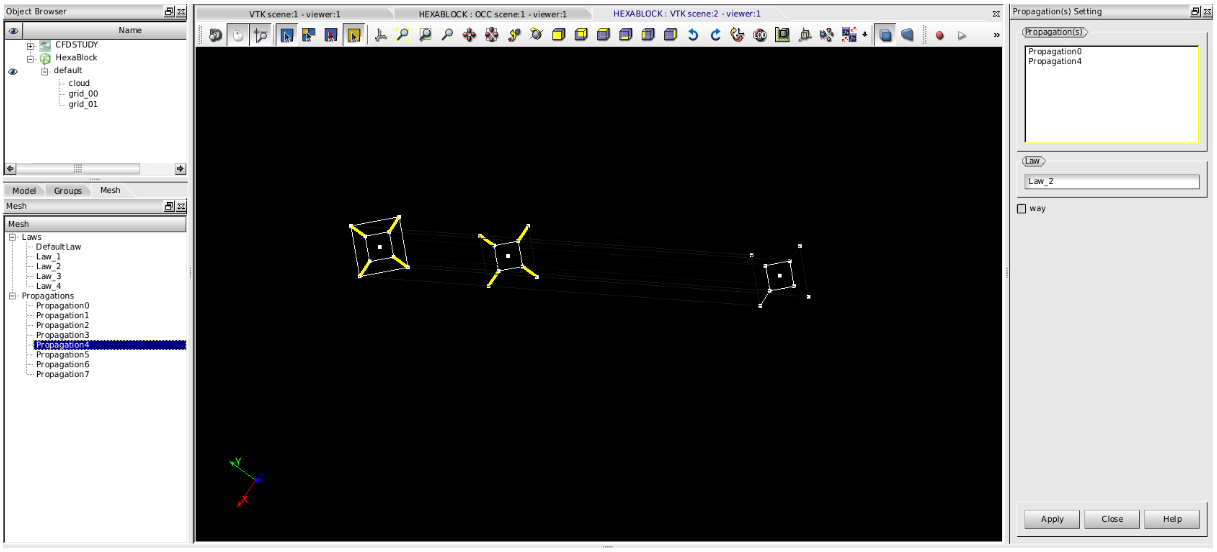
	Law			
	1	2	3	4
Number of nodes	8	4	100	422
Coefficient	1.0	1.0	0.97	1.0
Kind	Uniform	Uniform	Geometric	Uniform

Table III.1: Discretisation laws specification.

In order to associate a law to a propagation or a list of propagations, select ***Mesh⇒Set Propagation***. The associations between the laws and the propagations are shown in the Table III.2. Do not forget to press the key 'x' if you need to suppress one propagation selected.

	Propagation							
	0	1	2	3	4	5	6	7
Law	2	1	3	1	2	1	4	1

Table III.2: Propagations and associated Laws.



Propagation(s)
Propagation0
Propagation4

Law
Law_2

□ way

Figure III.6: Associating laws and propagations.

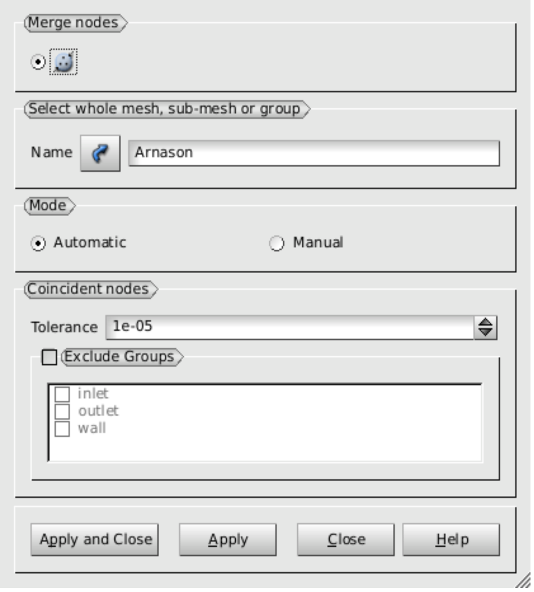
When all propagations have been associated to a law, you can create the mesh by selecting ***Mesh*⇒*Compute Mesh***. In the 'Compute Mesh' window, leave the dimension as 3, rename the mesh 'Arnason' and press 'Apply'. The mesh has been created in the Mesh module.

The volume mesh in each cylinder will have been generated by HexaBlock and each mesh will be conformal. However, at the domain origin, where the two cylinders are coincident, the meshes are not joined or merged automatically by HexaBlock. Hence there will be coincident or overlapping mesh vertices where the cylinders are coincident. These vertices must then be merged.

5 Merging Overlapping Vertices

The last step is to merge the overlapping vertices in SALOME. From the 'Mesh' module, open the mesh tree menu and display the mesh by clicking on the 'eye' icon next to 'Arnason'.

To merge the overlapping vertices, in the 'Mesh' module click on ***Modification*⇒*Transformation*⇒*Merge Nodes***. In the pop-up interface, leave all parameters at their default values and press 'Apply and Close' as shown in Figure III.7.



Merge nodes

○

Select whole mesh, sub-mesh or group

Name: Arnason

Mode

● Automatic ○ Manual

Coincident nodes

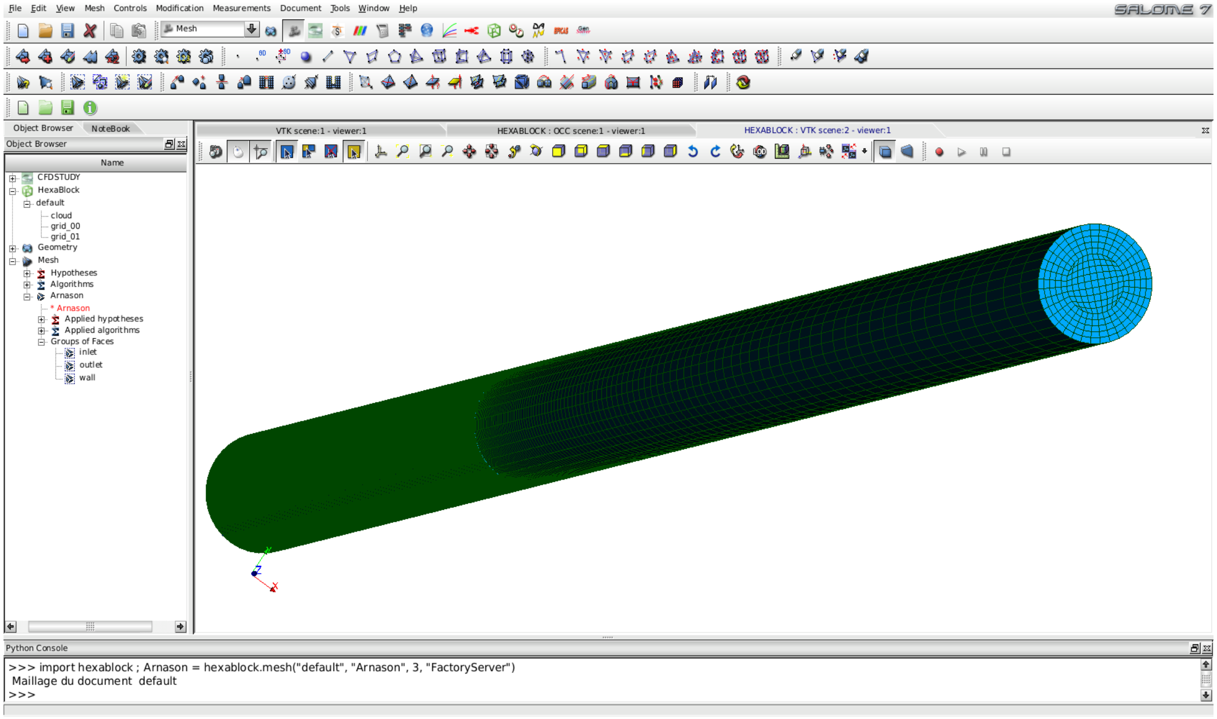
Tolerance: 1e-05

□ Exclude Groups

□ inlet
□ outlet
□ wall

Figure III.7: Merge Nodes.

The overlapping vertices will now be merged and a single volume mesh will be generated. Figure III.8 presents a zoomed view of the volume mesh at the domain origin.



```
>>> import hexablock ; Arnason = hexablock.mesh("default", "Arnason", 3, "FactoryServer")
 Maillage du document  default
>>>
```

Figure III.8: Volume mesh.

The generation of the computational domain and the volume mesh have now been completed. Save the SALOME file and export the mesh file in '.med' format by selecting from the main menu: ***File⇒Export⇒MED file***. For the file name, choose 'Mesh_ARNASON'; the '.med' extension is automatically added. You are now ready to move to the CFDStudy module in order to set up the CFD simulation.

Chapter IV

Part 3 - Single-Phase RANS Computation

1 What you Will Learn

In this third part of the tutorial you will learn how to set-up, run, and post-process the results of a steady-state single phase RANS calculation for the Arnason pipe generated in Section III. The setting up and running will be completed using the CFDStudy module in SALOME and you will also learn how to integrate user coding into a *Code_Saturne* calculation. The user coding will be used to specify the inlet boundary conditions. Probes will also be set in the computational domain and used in the analysis to verify that steady-state, converged results are obtained.

2 Setting up the CFD Simulation

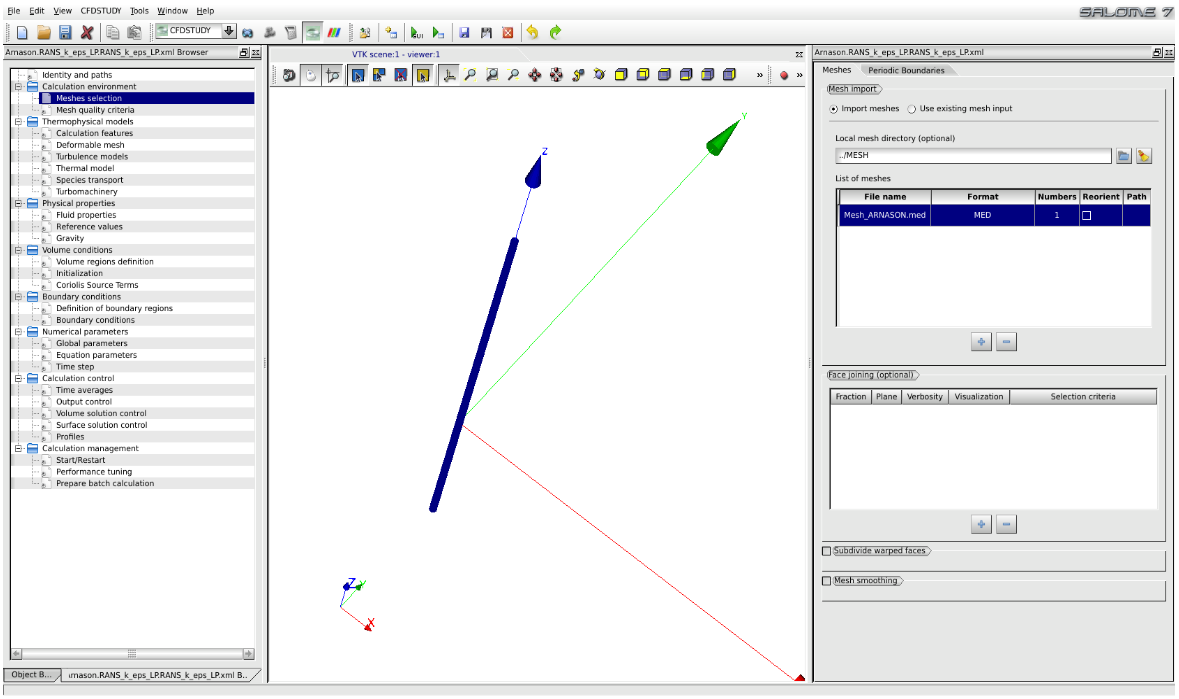
The CFD case is set-up and run from the CFDStudy module (Section 3).

In the CFDStudy module, create a 'New File' and verify that the case directory structure has been correctly recognised by clicking on the 'Identity and Paths' folder in the tree menu. If the case directory is correct you can continue. If not, you will need to set the correct directory. Then, save the file as 'RANS_k_eps_LP.xml'.

You can now proceed with setting up the case, following the top-down order of the folders in the left-hand column, starting with the mesh.

2.1 Selecting the Volume Mesh

Open the 'Calculation environment' folder and, in the 'Meshes' panel of the 'Meshes Selection', subfolder, add the mesh 'Mesh_ARNASON.med' to the initially empty list of meshes. This is done by clicking on the '+' icon in the panel and selecting the appropriate mesh in the MESH directory, as shown in Figure IV.1. No further input is necessary for the volume mesh.



Mesh import

● Import meshes ○ Use existing mesh input

Local mesh directory (optional)

../MESH

List of meshes

File name	Format	Numbers	Reorient	Path
Mesh_ARNASON.med	MED	1	□	

Face joining (optional)

Fraction	Plane	Verbosity	Visualization	Selection criteria

□ Subdivide warped faces

□ Mesh smoothing

Figure IV.1: Importing the mesh.

You can now go to the 'Thermophysical models' folder in order to specify the flow physics for the calculation.

2.2 Thermophysical Models

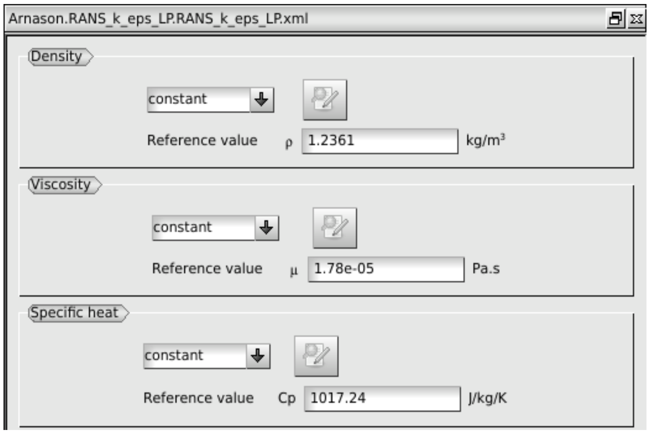
In the 'Calculation features' sub-folder, change the algorithm to 'steady flow' in the drop down menu. Leave all the other default values unchanged: 'multiphase flow', 'atmospheric flows', 'combustion', and the 'electrical' and 'compressible' models are all inactive.

In the 'Turbulence models' sub-folder, change 'Turbulence model' to 'k-epsilon Linear Production'. In the 'Advanced Options' sub-folder, ensure that the wall function type is set to 'Two scale model' and that 'Gravity terms in the turbulence equations' is selected.

No other settings are required for the thermo-physical models. You can now move to the 'Physical properties' folder.

2.3 Physical Properties

Given that the flow field is incompressible, the physical properties of the fluid are constant for a constant fluid temperature of $10°C$. Specify the value of each fluid property as shown in Figure IV.2.



Arnason.RANS_k_eps_LP.RANS_k_eps_LP.xml

Density

constant

Reference value ρ 1.2361 kg/m³

Viscosity

constant

Reference value μ 1.78e-05 Pa.s

Specific heat

constant

Reference value Cp 1017.24 J/kg/K

Figure IV.2: Specifying the fluid properties.

In the 'Gravity' sub-folder, set the acceleration of gravity by entering the value '$-9.81 m/s^2$' for its component in the vertical (Z) direction in the 'Gravity' panel.

No other settings are required in this folder. You can now move to the 'Volume conditions' folder.

2.4 Volume Conditions

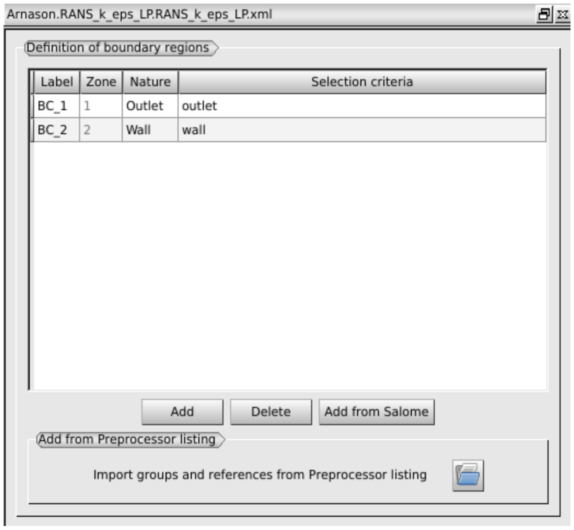
The initial values for the velocity are defined in the 'Initialization' tab of the 'Volume conditions' folder. The flow is initially stagnant by default.

No other settings are required in this folder. You can now move to the 'Boundary conditions' folder.

2.5 Boundary Conditions

Three boundary conditions are used in this study: inlet, wall and outlet. These conditions are listed in Table II.5. The inlet boundary condition will be defined is the 'cs_user_boundary_condition' subroutine so it does not need to be generated here (see 2.8).

First, create the 'wall' and 'outlet'boundary conditions: in the 'Boundary conditions' folder, select the 'Definition of boundary regions' sub-folder and click on the 'Add' button twice. Then, change the 'Selection criteria' name of each boundary to reflect the name of groups defined in the mesh, i.e. 'outlet' and 'wall' (Figure IV.3). Finally, change the type ('Nature') of the boundary to 'outlet' for the outlet boundary but leave the type of the wall boundary condition as 'wall'.


Arnason.RANS_k_eps_LP.RANS_k_eps_LP.xml
Definition of boundary regions
Add
Delete
Add from Salome
Add from Preprocessor listing
Import groups and references from Preprocessor listing

Label	Zone	Nature	Selection criteria
BC_1	1	Outlet	outlet
BC_2	2	Wall	wall

Figure IV.3: Defining the boundary conditions.

Having defined their type, the values to apply at each boundary could now be specified. However, as the default values are used in this tutorial, no further specification is necessary. To check the default values, select the 'Boundary conditions' sub-folder and click on the boundary of interest.

No other settings are required in this folder. You can now go to the 'Numerical parameters' folder.

2.6 Numerical Parameters

In the 'Numerical Parameters' folder and in the 'Global parameters' sub-folder, set the 'Velocity-Pressure algorithm' to SIMPLE. Leave as default all other settings.

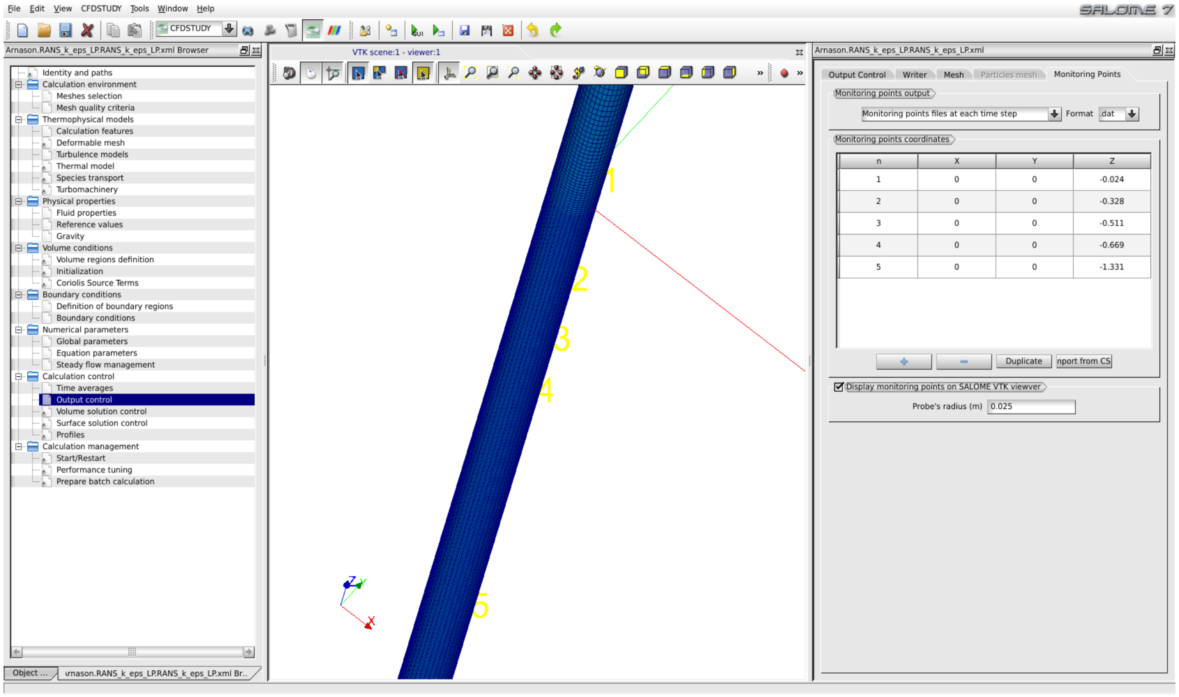
In the 'Equation parameters' sub-folder, the 'Solver' panel shows that pressure and velocity are solved for. In order to decrease overall computation time, in this instance it is possible to reduce the maximum iteration number to 1000 for each variable and to decrease the solver precision to 10^{-5} without having an impact on solution quality.

Leave as default the parameters in the 'Scheme' panel. However, if you have limited computational resources, you may change the second Centered scheme to the first order Upwind scheme for the convective variables.

In the 'Steady flow management' panel, keep the relaxation coefficient at 0.7 but set the number of iterations to 1000. For this tutorial, this number of iterations is sufficient for solution convergence and to achieve a steady flow solution.

2.7 Calculation Control

In the 'Calculation control' folder, select the 'Output control' sub-folder and go into the 'Monitoring Points' panel. Click on the '+' icon to add a probe, then enter the coordinate of the probe. Repeat this procedure for the probes of your choice. Monitoring probes can be useful to check the convergence of the simulation and it is recommend that monitoring points are specified along the axis of the pipe after the particle injection point. The monitoring points used for this tutorial are shown in Figure IV.4.



Monitoring points output

Monitoring points files at each time step Format .dat

Monitoring points coordinates

n	X	Y	Z
1	0	0	-0.024
2	0	0	-0.328
3	0	0	-0.511
4	0	0	-0.669
5	0	0	-1.331

☑ Display monitoring points on SALOME VTK viewver

Probe's radius (m) 0.025

Figure IV.4: Monitoring points.

The *Code_Saturne* calculation is now fully specified from the standpoint of the GUI and the *Code_Saturne* case 'xml' file should be saved. However, prior to running the simulation, the inlet boundary condition still needs to be specified. A parabolic law defining the V_z velocity component needs to be coded in the 'cs_user_boundary_conditions.f90' subroutine. This step is described next.

2.8 Programming the Inlet Boundary Condition with User Coding

To begin with, copy the sample file 'cs_user_boundary_conditions.f90' from the tutorial $/../MixingTee/RANS\_k\_eps\_LP/SRC/REFERENCE$ directory to your SRC directory. This is done in order to create a local copy which you will be able to customise and which will be automatically recompiled and linked to the 'cs_solver' executable at run time.

Once copied, open your local version of the file using the text editor of your choice. This subroutine contains several examples of different boundary conditions that can be used by *Code_Saturne*. In this tutorial, you will customise 'Example 1' with your own implementation of the V_z velocity as a function of radius (Eq. II.5). To keep your code clean, you may remove all the other examples from the file. The customised code is available with this tutorial and is already commented. Here we describe the main parts of this user coding and the logic behind them.

1. Declare your own local variables at the top of the subroutine, either as double precision real values or integer values
2. Initialise your own local variables
3. Use the subroutine 'getfbr' to select the faces attached to the 'inlet' boundary condition
4. Cycling through the boundary faces.
   (a) Apply the type 'ientre' to all boundary faces
   (b) Calculate the V_z velocity component and the turbulence based on the hydraulic diameter

3 Running and Analysing the Simulation

3.1 Running the Simulation

In the ‘Calculation management’ folder, click on the ‘Start/Restart’ sub-folder and check that the ‘calculation restart’ option is off.

In the panel of ‘Prepare batch calculation’ sub-folder,

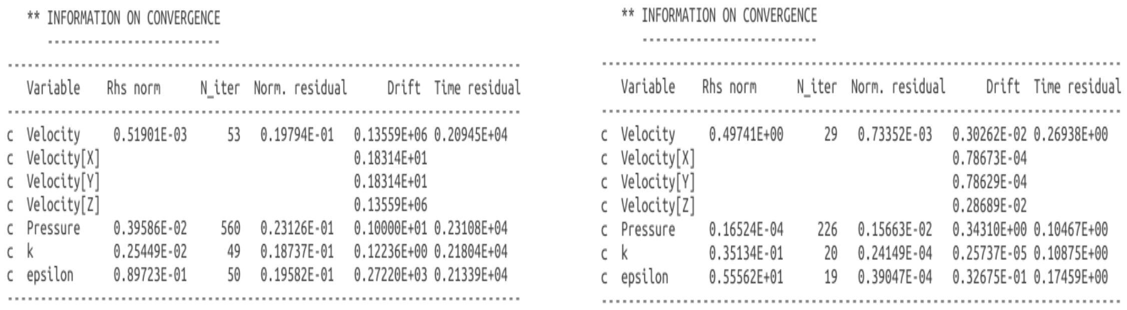
- ‘runcase’ for the ‘Script file’
- ‘standard’ (calculations without a prior step to import or process meshes or partitions)
- ‘1’ for the number of processors for the ‘Calculation script parameters’

are all set as default. You may increase the ‘Number of processes’ if you would like in order to run the simulation in parallel.

To run *Code_Saturne*, press the ‘Start calculation’ button. The pop-up panel for the run opens, listing in real time the different stages of the calculation, from user-subroutines compilation to saving the results.

3.2 Checking Calculation Convergence

Wait for the calculations to complete and open the ‘listing’ file in your ”ARNASON/RANS_k_eps_LP/RESU/DateOfRunTimeOfRun/” directory. Verify that the residuals listed under ‘derive’ in the ‘Information on Convergence’ table have dropped several orders of magnitude for all variables (pressure, velocity and temperature), showing that the calculations have fully converged to a steady-state solution (Figure IV.5).



```
   ** INFORMATION ON CONVERGENCE
      -------------------------

 ---------------------------------------------------------------------------
   Variable    Rhs norm      N_iter  Norm. residual      Drift  Time residual
 ---------------------------------------------------------------------------
 c  Velocity     0.51901E-03      53   0.19794E-01   0.13559E+06 0.20945E+04
 c  Velocity[X]                                      0.18314E+01
 c  Velocity[Y]                                      0.18314E+01
 c  Velocity[Z]                                      0.13559E+06
 c  Pressure     0.39586E-02     560   0.23126E-01   0.10000E+01 0.23108E+04
 c  k            0.25449E-02      49   0.18737E-01   0.12236E+00 0.21804E+04
 c  epsilon      0.89723E-01      50   0.19582E-01   0.27220E+03 0.21339E+04
 ---------------------------------------------------------------------------
```

(a) After 1 Iteration

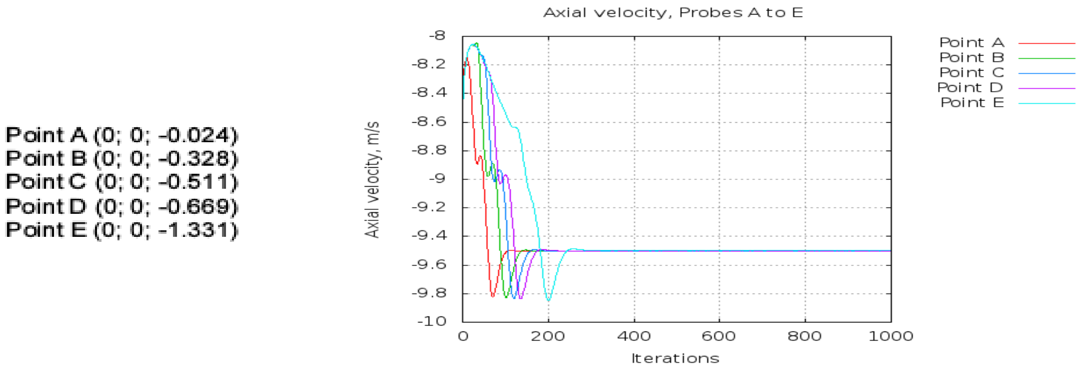
```
   ** INFORMATION ON CONVERGENCE
      -------------------------

 ---------------------------------------------------------------------------
   Variable    Rhs norm      N_iter  Norm. residual      Drift  Time residual
 ---------------------------------------------------------------------------
 c  Velocity     0.49741E+00      29   0.73352E-03   0.30262E-02 0.26938E+00
 c  Velocity[X]                                      0.78673E-04
 c  Velocity[Y]                                      0.78629E-04
 c  Velocity[Z]                                      0.28689E-02
 c  Pressure     0.16524E-04     226   0.15663E-02   0.34310E+00 0.10467E+00
 c  k            0.35134E-01      20   0.24149E-04   0.25737E-05 0.10875E+00
 c  epsilon      0.55562E+01      19   0.39047E-04   0.32675E-01 0.17459E+00
 ---------------------------------------------------------------------------
```

(b) After 1000 Iterations

Figure IV.5: Convergence history from the ‘listing’ file.

Further, check if the flow values are well-established and that the flow has reached a steady-state by plotting the values of velocity at the probes locations.

Figure IV.6 presents the evolution of the velocity at the five monitoring points placed along the pipe axis at the z coordinates -0.024, -0.328, -0.511, -0.669 and -1.331m. This figure indicates that, after an intial transient during which the flow develops from the initial solution, the flow is well established and steady within 400 iterations.



Point A (0; 0; -0.024)
Point B (0; 0; -0.328)
Point C (0; 0; -0.511)
Point D (0; 0; -0.669)
Point E (0; 0; -1.331)

(a) Monitoring point coordinates

(b) Evolution of axial velocity during the calculation

Figure IV.6: Monitoring the axial velocity

Having generated the steady-state, single phase RANS flow field on which the particles will be injected, the first step of the one-way coupling, two phase Lagrangian *Code_Saturne* modelling of Arnason's [4] experiments has now been completed. You can proceed to the setting-up, running, and analysis of the two phase Lagrangian *Code_Saturne* model in IV.

Chapter V

Part 4 - Two-Phase Lagrangian Computation

1 What you Will Learn

In this fourth part of this tutorial, you will learn how to set-up, run, and post-process a Lagrangian two-phase flow simulation in *Code_Saturne*, and how to compare the numerical results with the experimental data of [4].

2 Setting up the Lagrangian Simulation

In this section, only the modifications that need to be applied to the single phase flow set-up for the Lagrangian calculation are discussed here. Everything else remains as described in IV. The information added for the Lagrangian particulates include the specification of the injection point for a particle size of $5\mu m$ diameter.

Right click on the 'RANS_k_eps_LP_5UM' case which you created earlier (3) in the tree of the CFD-Study module and select "Launch GUI". Create a new file by clicking on the 'New file' icon and verify that the the case directory structure has correctly been recognised. If so, save the file as 'RANS_k_eps_LP_5UM.xml', if not change the directory structure so that it is correct and then save the 'RANS_k_eps_LP_5UM.xml' file.

You can now set-up the Lagrangian two-phase flow case.

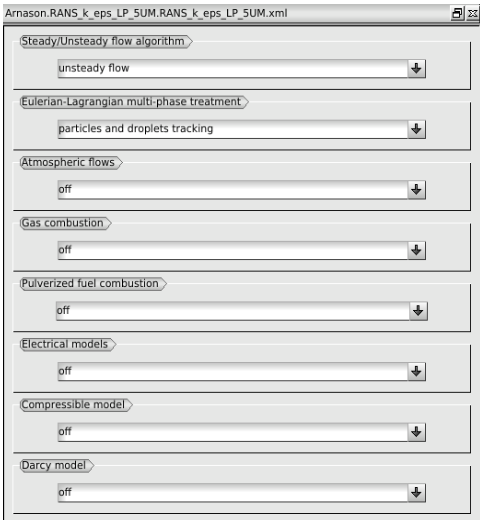
2.1 Selecting the Volume Mesh

In the 'Meshes' panel of the 'Meshes Selection' folder, ensure that the 'Mesh_ARNASON.med' is in the list. If not, add it to the list by clicking on the '+' icon and searching for it in the 'Arnason_LAG/MESH' directory

2.2 Setting up the Lagrangian Model

Calculation Features

In the panel of the 'Calculation features' folder, change the 'Steady/Unsteady flow algorithm' to 'unsteady flow' in the drop down menu. In the same folder, select 'particles and droplets tracking' in the drop down menu of the 'Eulerian-Lagrangian multi-phase treatment' menu as shown in Figure V.1. Ensure that 'atmospheric flows', 'combustion' and the 'electrical' and 'compressible' models are all 'off'



Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml

Steady/Unsteady flow algorithm: unsteady flow

Eulerian-Lagrangian multi-phase treatment: particles and droplets tracking

Atmospheric flows: off

Gas combustion: off

Pulverized fuel combustion: off

Electrical models: off

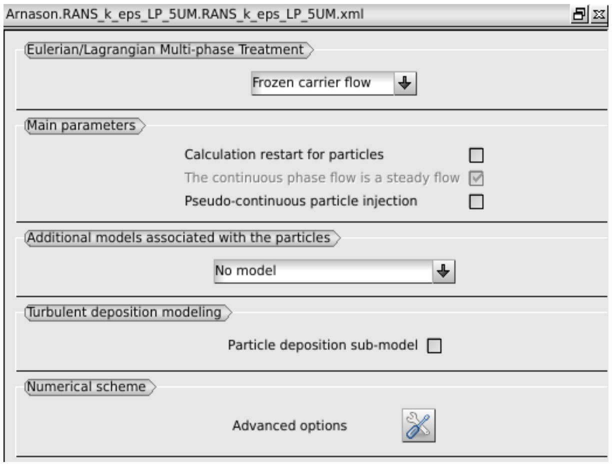
Compressible model: off

Darcy model: off

Figure V.1: Selecting the flow physics.

Particles and Droplet Tracking

In the ‘Particles and droplets tracking’ folder that has now appeared in the GUI tree menu, click on the sub-folder ‘Global settings’ to display the panel. For the ‘Eulerian/Lagrangian Multi-phase Treatment’, select the option ‘Frozen carrier flow’ in the drop down menu, as shown in Figure V.2. In choosing this option, the box ‘The continuous phase flow is a steady flow’ is automatically ticked. Leave all other settings in this panel at their default options.



Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml

Eulerian/Lagrangian Multi-phase Treatment

Frozen carrier flow

Main parameters

Calculation restart for particles □

The continuous phase flow is a steady flow ☑

Pseudo-continuous particle injection □

Additional models associated with the particles

No model

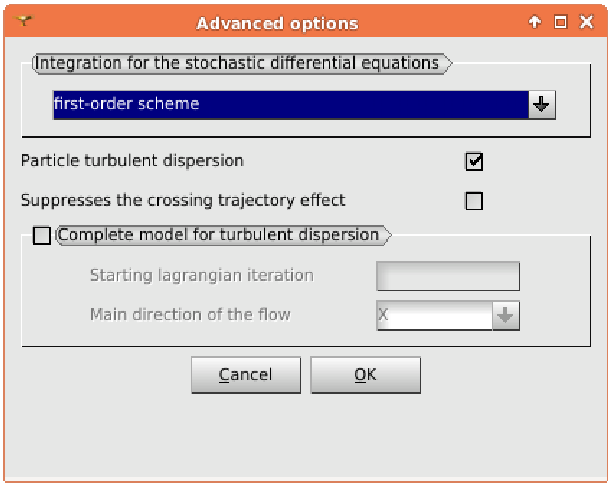
Turbulent deposition modeling

Particle deposition sub-model □

Numerical scheme

Figure V.2: Global settings menu.

Click on the ‘Advanced options’ icon to specify the numerical scheme, as shown in Figure V.3. Choose the ‘first order scheme’ for the ‘integration for the stochastic differential equations’ option and tick the ‘particle turbulent dispersion’ box.



Advanced options

Integration for the stochastic differential equations

first-order scheme

Particle turbulent dispersion ☑

Suppresses the crossing trajectory effect □

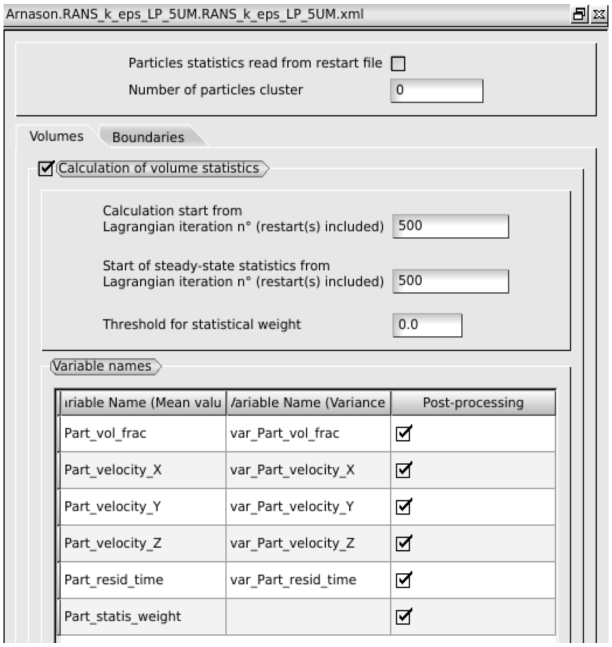
□ Complete model for turbulent dispersion

Starting lagrangian iteration

Main direction of the flow X

Figure V.3: Advanced options of the numerical scheme menu.

In the ‘Statistics’ sub-folder, select all parameters, as presented in Figure V.4. The number of particles present in the computational domain is constant after 200 iterations, hence the statistics are started after iteration 500 (cf. Figure VI.1 in 3).



Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml

Particles statistics read from restart file □

Number of particles cluster 0

Volumes | Boundaries

☑ Calculation of volume statistics

Calculation start from Lagrangian iteration n° (restart(s) included) 500

Start of steady-state statistics from Lagrangian iteration n° (restart(s) included) 500

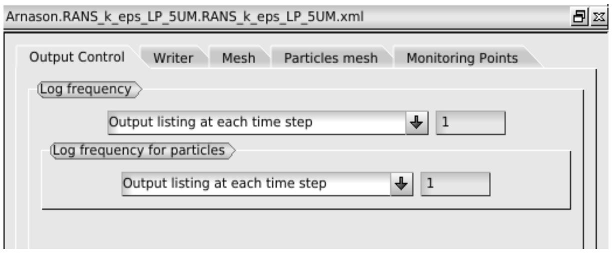
Threshold for statistical weight 0.0

Variable names

ıriable Name (Mean valu	/ariable Name (Variance	Post-processing
Part_vol_frac	var_Part_vol_frac	☑
Part_velocity_X	var_Part_velocity_X	☑
Part_velocity_Y	var_Part_velocity_Y	☑
Part_velocity_Z	var_Part_velocity_Z	☑
Part_resid_time	var_Part_resid_time	☑
Part_statis_weight		☑

Figure V.4: Statistics menu.

In the panel of the 'Output' sub-folder, ensure that the 'Output listing at each time step' is set to 1. This will output particle information at every time step. Next, tick the boxes for 'Trajectory mode' and 'Displacement Mode' and set the 'Output frequency' to 1. Select all the options for the different variables to save. The final set-up for this panel is shown in Figure V.5 and Figure V.6.



Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml

Output Control | Writer | Mesh | Particles mesh | Monitoring Points

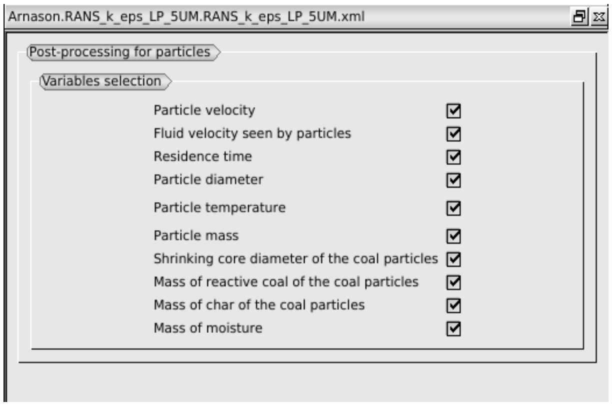
Log frequency

Output listing at each time step 1

Log frequency for particles

Output listing at each time step 1

Figure V.5: Output menu.



Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml

Post-processing for particles

Variables selection

Particle velocity ☑

Fluid velocity seen by particles ☑

Residence time ☑

Particle diameter ☑

Particle temperature ☑

Particle mass ☑

Shrinking core diameter of the coal particles ☑

Mass of reactive coal of the coal particles ☑

Mass of char of the coal particles ☑

Mass of moisture ☑

Figure V.6: Solution controls.

Particle Boundary Conditions

The next step is to set the 'Particles boundary conditions', which specify the boundary conditions for the particulate field.

To start with, create the 'wall' and 'outlet' boundaries as you did previous in 2.5. However, this time, also add the 'inlet' boundary in the 'Definition of boundary region' sub-folder, but do not specify the boundaries further in this folder. These will be specified for the particulates in the specialised, 'Particles boundary conditions' sub-folder.

Go to the 'Particles boundary conditions' sub-folder and change the type to 'Particle-boundary interaction' for the three boundary conditions: inlet, outlet and wall. In turn, select the following boundary conditions for the different boundaries:

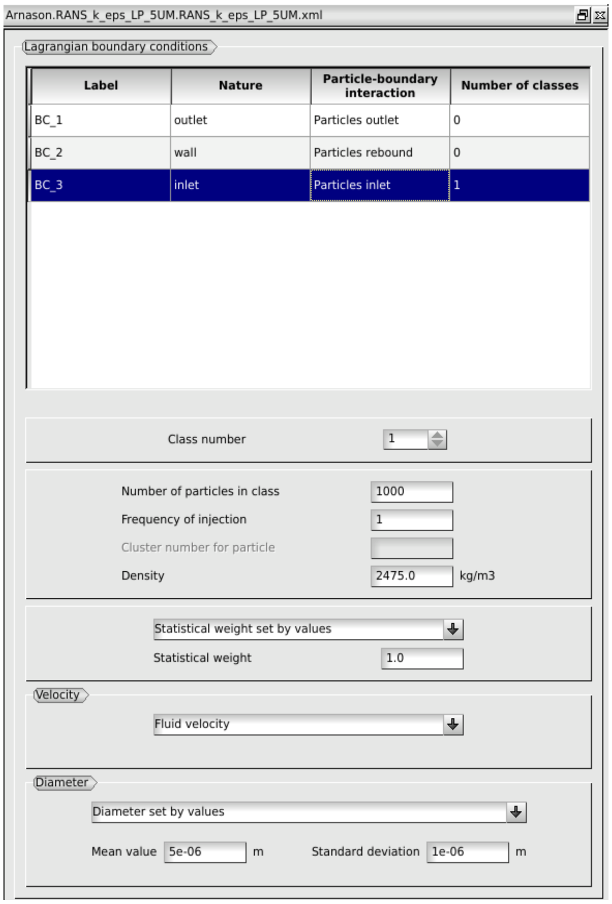
- 'outlet': 'Particles outlets'
- 'wall': 'Particles rebound' ⇒ particles bounce off walls without loss of energy
- 'inlet': 'Particles inlet'

Complete the 'inlet' boundary condition by defining the particles which will be injected at the inlet plane. As you are only simulating one type of particles, the glass beads $5\mu m$ in mean diameter, enter '1' as number of classes for the 'inlet' and specify the:

- number of particles injected
- frequency
- density
- injection velocity
- mean diameter
- standard deviation

corresponding to the experiment (1.5).

The final set-up for this panel is shown in Figure V.7.



Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml

Lagrangian boundary conditions

Label	Nature	Particle-boundary interaction	Number of classes
BC_1	outlet	Particles outlet	0
BC_2	wall	Particles rebound	0
BC_3	inlet	Particles inlet	1

Class number: 1

Number of particles in class: 1000

Frequency of injection: 1

Cluster number for particle:

Density: 2475.0 kg/m3

Statistical weight set by values

Statistical weight: 1.0

Velocity

Fluid velocity

Diameter

Diameter set by values

Mean value: 5e-06 m   Standard deviation: 1e-06 m

Figure V.7: Particles boundary conditions panel.

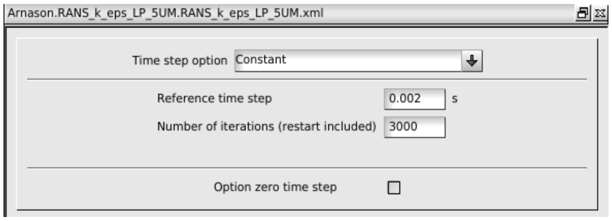
So far, the particles are set-up to be injected from the inlet boundary condition defined in the 'HexaBlock' module of SALOME and placed at the entrance to the pipe. However, in the experiments, the particles are injected further downstream of the pipe's inlet plane. Therefore, the coordinates of the injection point of the particles must be modified to match the experiments. This will done by programming them in the user subroutine *USLAIN* of user file *cs_user_particle_traking.f90* (3), as detailed below 3.1.

This concludes the set-up of the specifics of the Lagrangian two-phase mode. To complete the model in the GUI before moving to the required user coding, the numerical parameters should now be specified.

2.3 Numerical Parameters

In the 'Numerical Parameters' folder, leave as default the settings in the 'Global parameters' sub-folder but select 'SIMPLEC' as the 'Velocity-Pressure algorithm'. Leave as default the parameters in the 'Equation parameters' folder.

In the 'Time step' panel choose the 'constant' option for the time step and specify a reference time step of 0.002s with 3000 iterations. This will run the two-phase flow calculation for 2000 iterations, given that a *Code_Saturne* restart incudes the number of iterations previously completed. The final set-up for this panel is shown in Figure V.8.



Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml

Time step option: Constant

Reference time step: 0.002 s

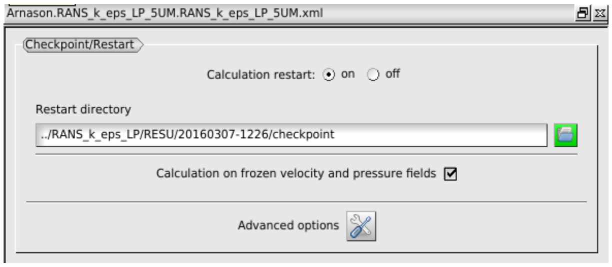
Number of iterations (restart included): 3000

Option zero time step □

Figure V.8: Time step menu.

2.4 Calculation Management

In the 'Calculation management' folder, go to the 'Start/Restart' tab to choose the restart file of the single phase fluid calculation. Then tick the 'Calculation on frozen velocity and pressure fields' box as shown in Figure V.9, so that the particle field is injected on top of the previously calculated single-phase field.



Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml

Checkpoint/Restart

Calculation restart: ● on ○ off

Restart directory

../RANS_k_eps_LP/RESU/20160307-1226/checkpoint

Calculation on frozen velocity and pressure fields ☑

Advanced options

Figure V.9: Start/restart menu.

The set-up of the two-phase flow model in the GUI is now complete. You should now save the 'RANS_k_eps_LP_5UM.xml' file. However, before you can run the simulation, user subroutines *uslain* and *uslast* must be programed in order to change the injection surface to an injection point and to add output statistics concerning the particle concentrations and the particle axial and radial velocities at the experimental measurement planes.

3 Programming the User Coding

Copy the sample file *cs_user_particle_tracking.f90* from the tutorial directory /../ARNASON/RANS_k_eps_LP_5UM/SRC/REFERENCE to your SRC directory in order to create a local copy. This local copy can be customised and will automatically be compiled and linked to the 'cs_solver' executable at run time.

3.1 *USLAIN* User Subroutine

Open your local version of the particle tracking Fortran file using the text editor of your choice. The specification of the injection boundary conditions for the particles is done in the *uslain* subroutine. Currently, the particles are injected at the 'inlet' boundary of the computational domain. This is not correct and the particle injection coordinates need to be modified in order that particle injection takes place at the origin of the flow domain.

The customised code is provided with this tutorial which can be used directly or used as a working example. Here we describe the main parts of this code and the logic behind them.

1. Cycling through the new particles injected in the computational domain:

   (a) Modify the coordinates of all particles in order to inject them at the domain origin and not the inlet boundary
   (b) Update the fluid velocity seen by the particles and the particle velocity by the fluid velocity at the domain origin
   (c) Clip the diameter of the particles in order to remove any unreasonable diameters

2. Activate the part concerning the simulation of the instantaneous turbulent fluid flow velocities seen by the solid particles along their trajectories

3.2 *USLAST* User Subroutine

The *uslast* subroutine is used to generate additional output data relating to the particles and is also coded in the *cs_user_particle_tracking.f90* file. The customised code is provided with this tutorial which can be used directly or used as a working example. Here we describe the main parts of this code and the logic behind them:

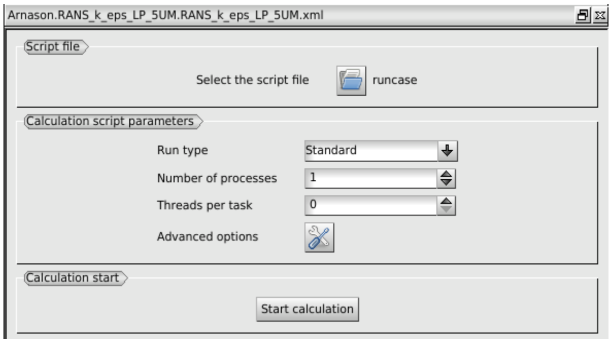
1. Activate 'Example 4' in 'uslast' by setting the flag to 'True' from 'False'
2. Specify the four measurement planes where the Lagrangian statistics will be calculated
3. Give a name to the four files that will be used by the subroutine to export particle data
4. Initialise the 'mean dispersed phase velocity' and the 'dispersed phase volumetric concentration' parameters which are required to extract particle data
5. Cycle across all cells of each measurement plane to extract the particle concentrations and the particle radial and axial velocities

The case is now ready to run.

4 Running and Analysing the Simulation

In the 'Calculation management' folder go to the panel of the 'Prepare batch calculation' sub-folder. In this panel (Figure V.10), keep all the default selections:

- 'runcase' for the 'Script file'
- 'standard' (calculations without a prior step to import or process meshes or partitions)
- '1' for the number of processors for the 'Calculation script parameters'


Arnason.RANS_k_eps_LP_5UM.RANS_k_eps_LP_5UM.xml
Script file
Calculation script parameters
Advanced options
Calculation start
Start calculation

Select the script file runcase

Run type Standard

Number of processes 1

Threads per task 0

Figure V.10: Batch calculation settings.

Press the 'Start calculation' button to run *Code_Saturne*. The pop-up panel for the run opens, listing in real time the different stages of the calculation, from compilation of the user-subroutines to the saving of the results.

5 Post-processing the Results

For the post-processing of the results, move to the 'ParaVis' module. In the 'Pipeline Browser' panel on the left-hand side, right click and select 'Open' in the drop-down menu. Point to the 'RESULTS.case' file in the RESU directory for the run that has just finished:

'/../ARNASON_LAG/RANS_k_eps_LP_5UM/RESU/DateOfRunTimeOfRun/postprocessing/RESULTS .case'

and to the 'PARTICLES.case' in:

'/../ARNASON_LAG/RANS_k_eps_LP_5UM/RESU/DateOfRunTimeOfRun/postprocessing /PARTICLES.case.'

Follow the steps described in tutorials [5, 7], to create the 'ExtractBlock' and 'CellDataPointData' objects, and post-process the results.

Chapter VI

Part 5 - Results

1 What you Will Learn

In this final part of the tutorial, you will learn how to analyse in detail the calculated particle field and compare it to the single-phase field, and to compare the numerical and experimental results using the particle output file that you set-up in the *cs_user_particle_tracking.f90* user routine.

2 Verifying the Simulation

During the Lagrangian calculation, a 'listla' file is automatically by created *Code_Saturne* containing the data listed in Table VI.1:

Column	Description
1	Number of the time step
2	Lagrangian physical time
3	The number of instantaneous particles in the domain
4	The number of instantaneous particles in the domain (with weighting)
5	The number of particles injected in the domain
6	The number of particles injected in the domain (with weight)
7	Instantaneous number of particles leaving the domain, or deposed and eliminated
8	Instantaneous number of particles leaving the domain, sticking to the wall and eliminated(with weighting)
9	Instantaneous number of particles sticking to the wall
10	Instantaneous number of particles sticking to walls (with weighting)
11	Instantaneous number of particles lost
12	Instantaneous number of particles lost (with weighting)
13	Percentages of particles lost

Table VI.1: Description of the data in the 'listla' file..

This information can be used to evaluate the convergence of the simulations.

For example, Figure VI.1 and Figure VI.2 present, respectively, the number of particles in the domain and the number of particles leaving the domain during the Lagrangian simulation. It can be seen that both the number of particles in the domain and the number of particles leaving the domain is well established and remains stable after 250 time steps. The earlier decision, when setting up the Lagrangian model, of starting the particle statistical analysis at the 500^{th} time step is validated by this data analysis.

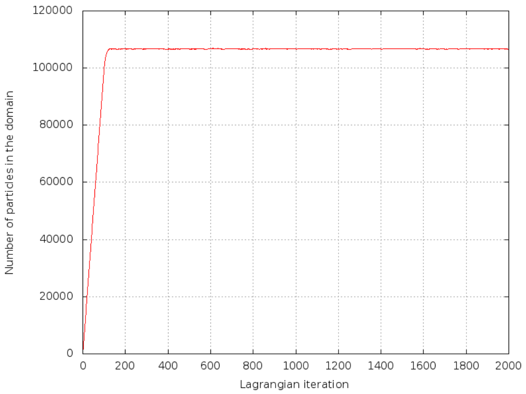


Figure VI.1: Number of particles in the computational domain.

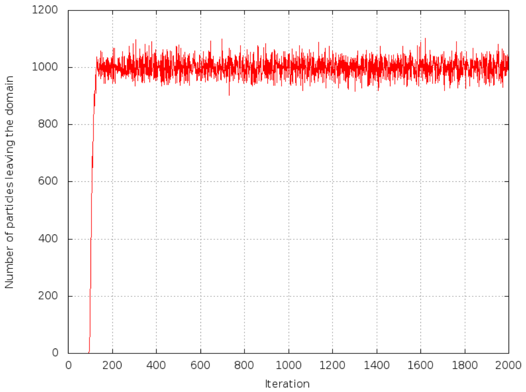


Figure VI.2: Number of particles leaving the computational domain.

3 Flow Field

Starting with the analysis of the flow field in ParaVis, Figure VI.3 presents a countour plot of the particle volume fraction in a 2D plane along the centre line of the pipe. It can be seen that the majority of the particles injected into the flow domain remain along or near the pipe axis.

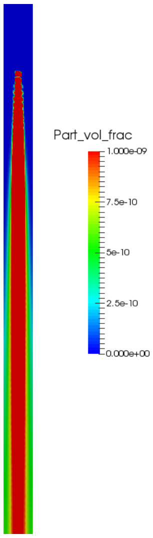
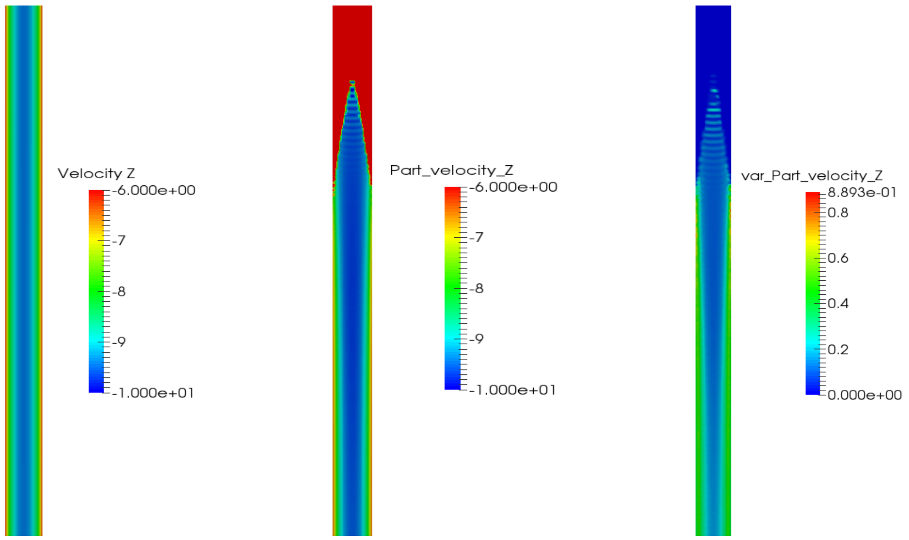


Figure VI.3: Volume fraction of the particles in the pipe.

Figure VI.4 presents the V_z velocity component of the carrier fluid, the vertical velocity of the particles and the V_z velocity variance of these particles, also on 2D slices in the yz plane along the pipe centre line.



(a) V_z velocity of the fluid (b) V_z velocity of the particles (c) Variance of the V_z velocity of the particles

Figure VI.4: Visualization in the yz plane along the pipe axis

It can be seen that, as the particles are very small, they are entrained by the fluid at the fluid velocity, with both the carrier fluid and the particles achieving a maximum velocity of the order of -9.5m/s, Figure VI.4a and Figure VI.4b.

The particle response and flow time scales may be compared to verify that, in this instance, the particles are expected to closely follow the carrier fluid. For this low particle-Reynolds number flow, the relaxation time, τ_p, of the particle is given by:

$$\tau_p \approx \frac{\rho_p d_p^2}{18\mu} = 1.92 \times 10^{-4} s \tag{VI.1}$$

For turbulent dispersion, the flow time scale may be estimated as the characteristic time of the turbulence,τ_{12}^t, calculated at the injection point:

$$\tau_{12}^t = \frac{3}{2} C_\mu \frac{k^2}{\epsilon} = 2.726 \times 10^{-3} s \tag{VI.2}$$

Therefore,$\frac{\tau_p}{\tau_{12}^t} \ll 1.0$, which confirms that the particles will follow the carrier fluid turbulence.

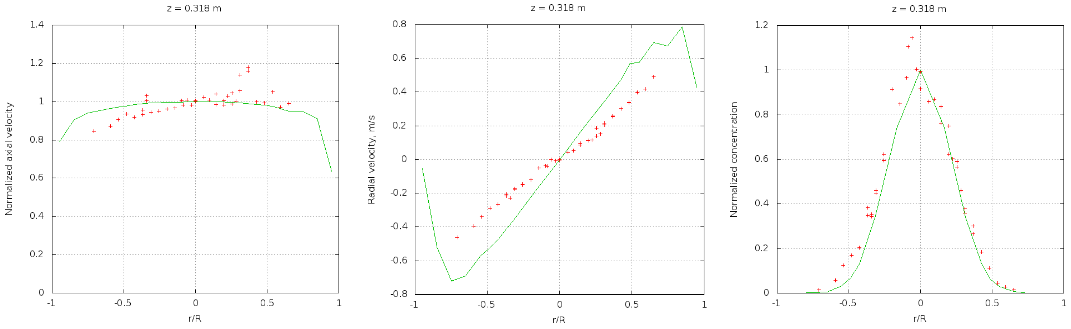
The variance of the vertical velocity of the particles (Figure VI.4c) can be seen to be at a minimum along the pipe axis and at its highest close to the flow domain wall, due to the near wall effects such as the boundary layer and the particle rebound condition.

4 Comparison of Predicted and Experimental Data

The numerical and experimental data can be compared using the output data files 'Z318', 'Z502', 'Z679', 'Z132' which *Code_Saturne* wrote at the end of the calculation as a result of the programming in subroutine *uslast* (3.2).

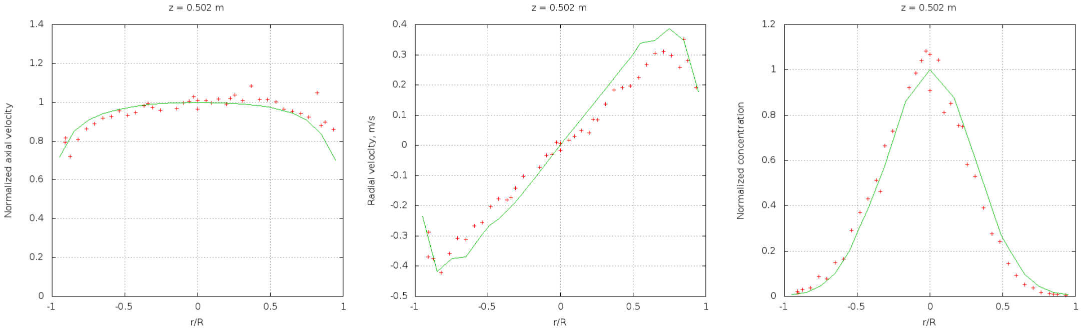
These files contain the particle normalised axial velocity, the particle concentration and the particle radial velocity at the $z = 0,318$, $z = 0,502$, $z = 0,679$ and $z = 1,32m$ planes where experimental data is also available. For convenience, the experimental data at these locations has been reproduced in Appendix 1 from [4].

Figure VI.5 to Figure VI.8 present the predicted (green line) and measured (red symbols) data at the different measuring planes. The figures show that the calculated values of the axial velocity and particle concentration are in good agreement with the experimental data for all measurement planes. The radial component of the velocity is also in good agreement with the experiment data, except for z = 1.32m. As the radial velocity decreases with the distance from the injection point and the concentration of particles near the walls increases, the error in the numerical results increases.



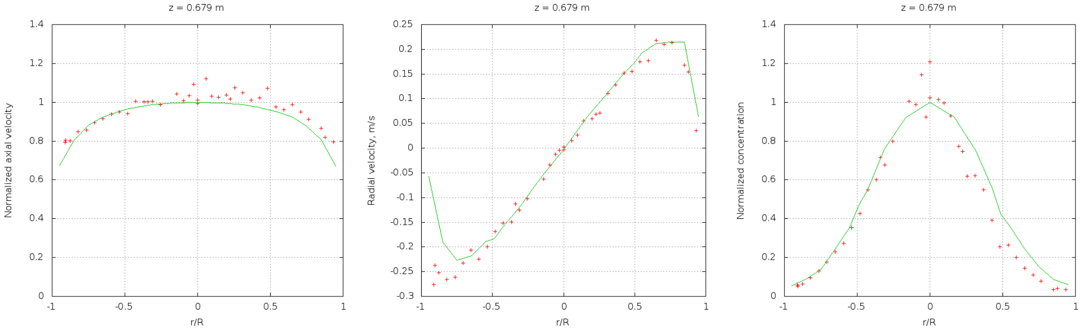
(a) Axial velocity (b) Radial velocity (c) Particle concentration

Figure VI.5: Numerical (line) and experimental[4] (symbols) results at z = 0.318m.



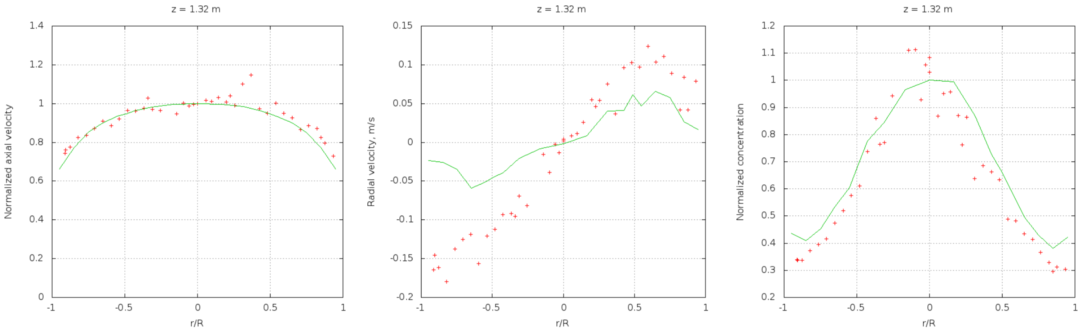
(a) Axial velocity. (b) Radial velocity. (c) Particle concentration

Figure VI.6: Numerical (line) and experimental[4] (symbols) results at z = 0.502m.



(a) Axial velocity. (b) Radial velocity. (c) Particle concentration

Figure VI.7: Numerical (line) and experimental[4] (symbols) results at z = 0.679m.



(a) Axial velocity.

(b) Radial velocity.

(c) Particle concentration

Figure VI.8: Numerical (line) and experimental[4] (symbols) results at z = 1.32m.

Chapter VII

References

1 References

[1] www.salome-platform.org

[2] F. Archambeau, N. Méchitoua, M. Sakiz,
Code_Saturne: a Finite Volume Code for the Computation of Turbulent Incompressible Flows - Industrial Applications,
International Journal on Finite Volumes, Vol. 1, 2004.

[3] www.code-saturne.org

[4] G. Arnason,
Measurement of particle dispersion in turbulent pipe flow,
Washington State University, Department of Mechanical Engineering, 1982

[5] EDF,
Tutorial 1: Shear Driven Cavity Flow,
Code_Saturne Tutorial Series

[6] http://docs.salome-platform.org/salome_6_6_0/gui/HEXABLOCK/index.html

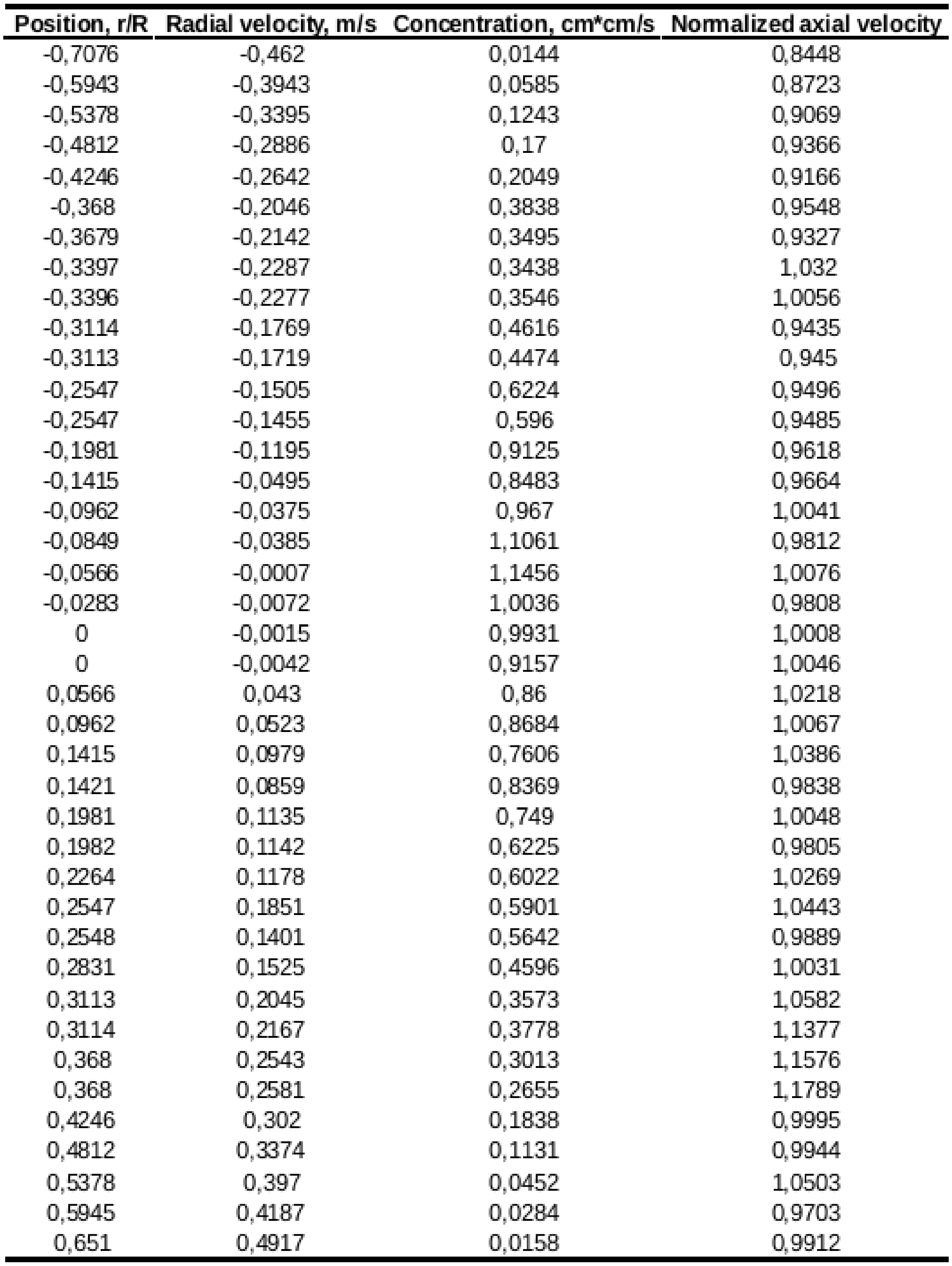
[7] EDF,
Tutorial 3: Heated Square Cavity Flow,
Code_Saturne Tutorial Series

Chapter VIII

Appendix

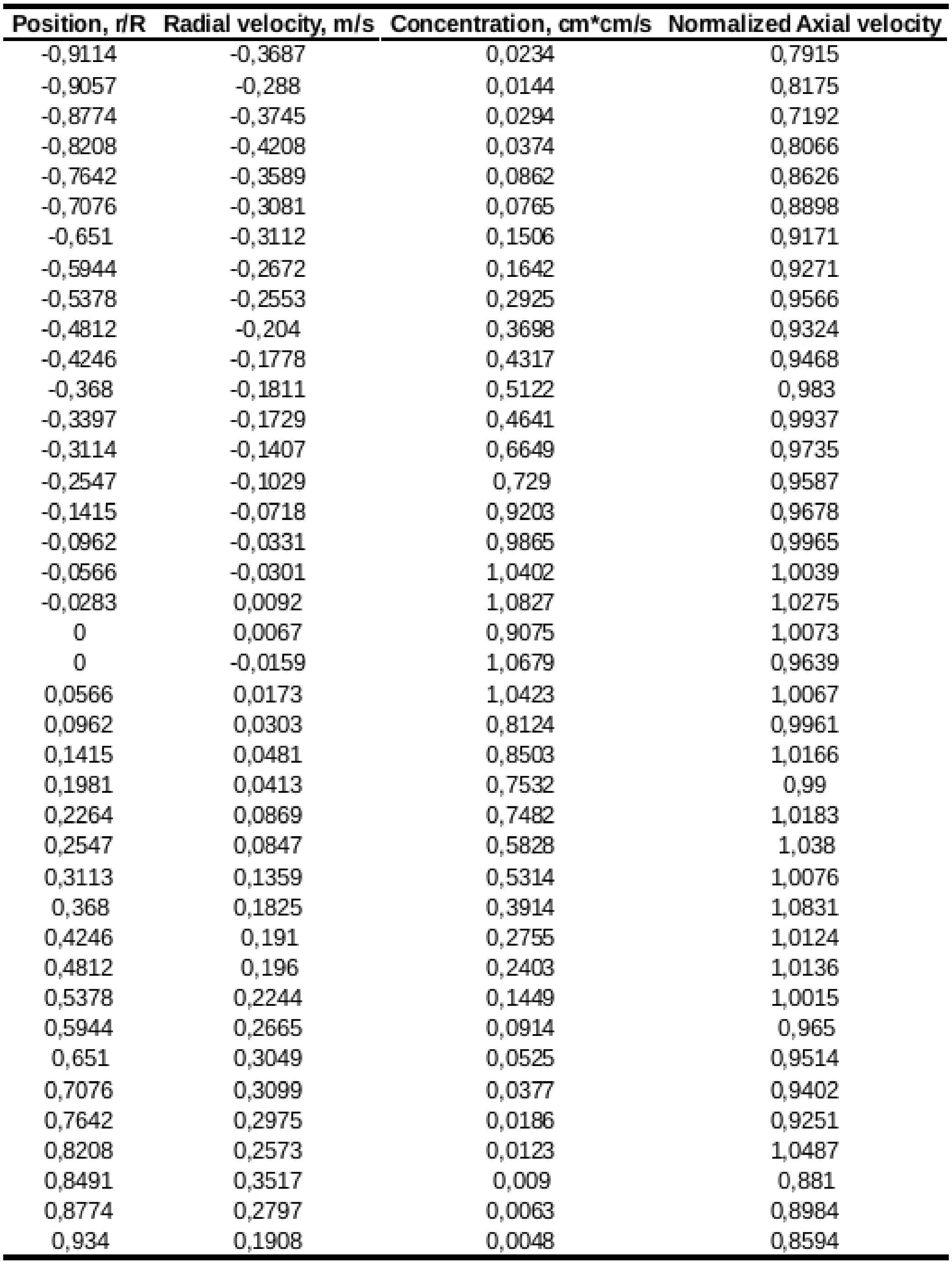
1 Appendix A – Experimental Data from [4]

The experimental data from ARNASON [4], for the particles of 5μm diameter are presented in this appendix. Experimental data of particle radial velocity, concentration and normalized axial velocity for the normalised radius at z = 0.318, 0.502, 0.679 and 1.32m are listed in the Table VIII.1 to VIII.4 respectively.



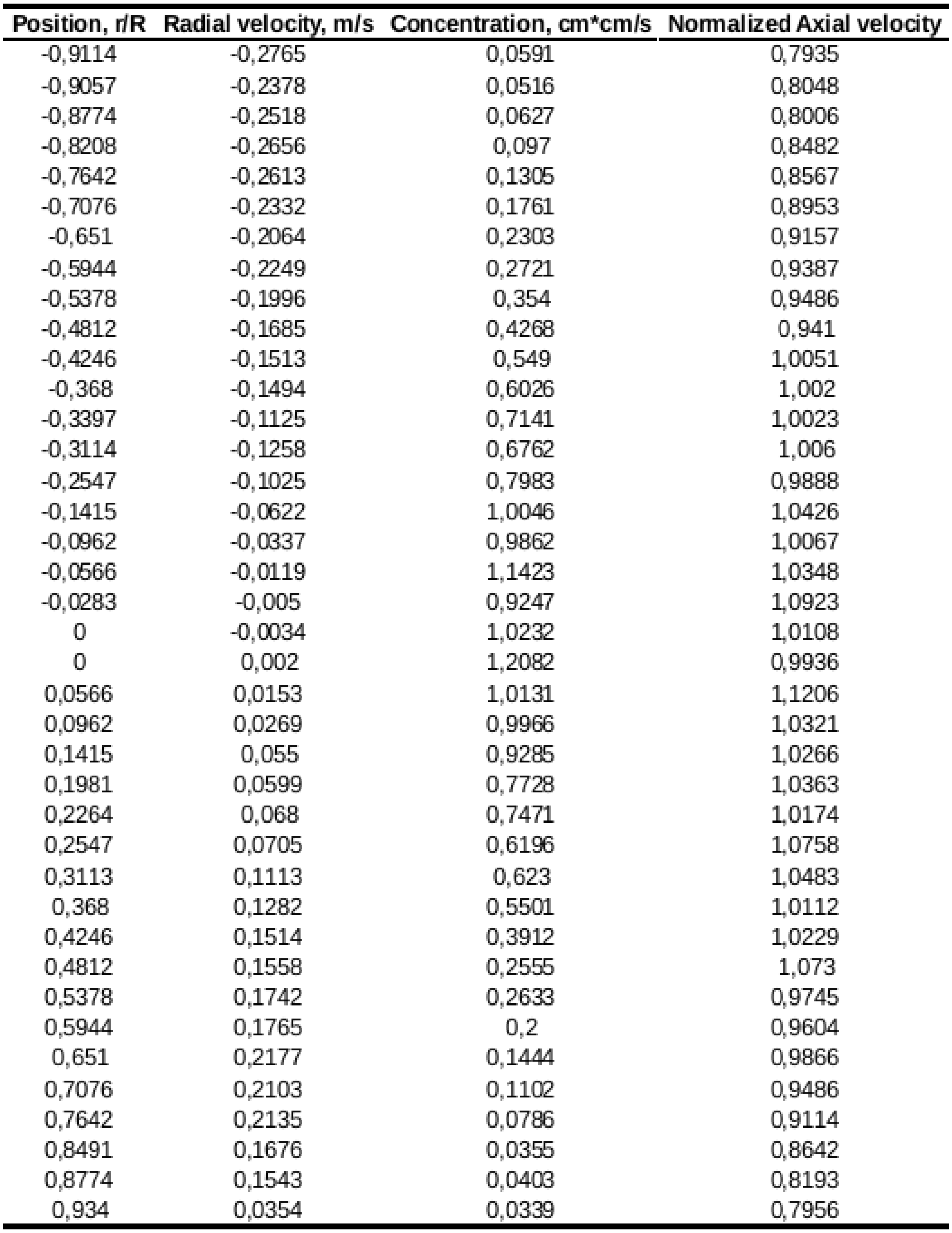
Position, r/R	Radial velocity, m/s	Concentration, cm*cm/s	Normalized axial velocity
-0,7076	-0,462	0,0144	0,8448
-0,5943	-0,3943	0,0585	0,8723
-0,5378	-0,3395	0,1243	0,9069
-0,4812	-0,2886	0,17	0,9366
-0,4246	-0,2642	0,2049	0,9166
-0,368	-0,2046	0,3838	0,9548
-0,3679	-0,2142	0,3495	0,9327
-0,3397	-0,2287	0,3438	1,032
-0,3396	-0,2277	0,3546	1,0056
-0,3114	-0,1769	0,4616	0,9435
-0,3113	-0,1719	0,4474	0,945
-0,2547	-0,1505	0,6224	0,9496
-0,2547	-0,1455	0,596	0,9485
-0,1981	-0,1195	0,9125	0,9618
-0,1415	-0,0495	0,8483	0,9664
-0,0962	-0,0375	0,967	1,0041
-0,0849	-0,0385	1,1061	0,9812
-0,0566	-0,0007	1,1456	1,0076
-0,0283	-0,0072	1,0036	0,9808
0	-0,0015	0,9931	1,0008
0	-0,0042	0,9157	1,0046
0,0566	0,043	0,86	1,0218
0,0962	0,0523	0,8684	1,0067
0,1415	0,0979	0,7606	1,0386
0,1421	0,0859	0,8369	0,9838
0,1981	0,1135	0,749	1,0048
0,1982	0,1142	0,6225	0,9805
0,2264	0,1178	0,6022	1,0269
0,2547	0,1851	0,5901	1,0443
0,2548	0,1401	0,5642	0,9889
0,2831	0,1525	0,4596	1,0031
0,3113	0,2045	0,3573	1,0582
0,3114	0,2167	0,3778	1,1377
0,368	0,2543	0,3013	1,1576
0,368	0,2581	0,2655	1,1789
0,4246	0,302	0,1838	0,9995
0,4812	0,3374	0,1131	0,9944
0,5378	0,397	0,0452	1,0503
0,5945	0,4187	0,0284	0,9703
0,651	0,4917	0,0158	0,9912

Table VIII.1: Experimental data from ARNASON obtained for particles of 5μm [4] at $z = 0,318$m.



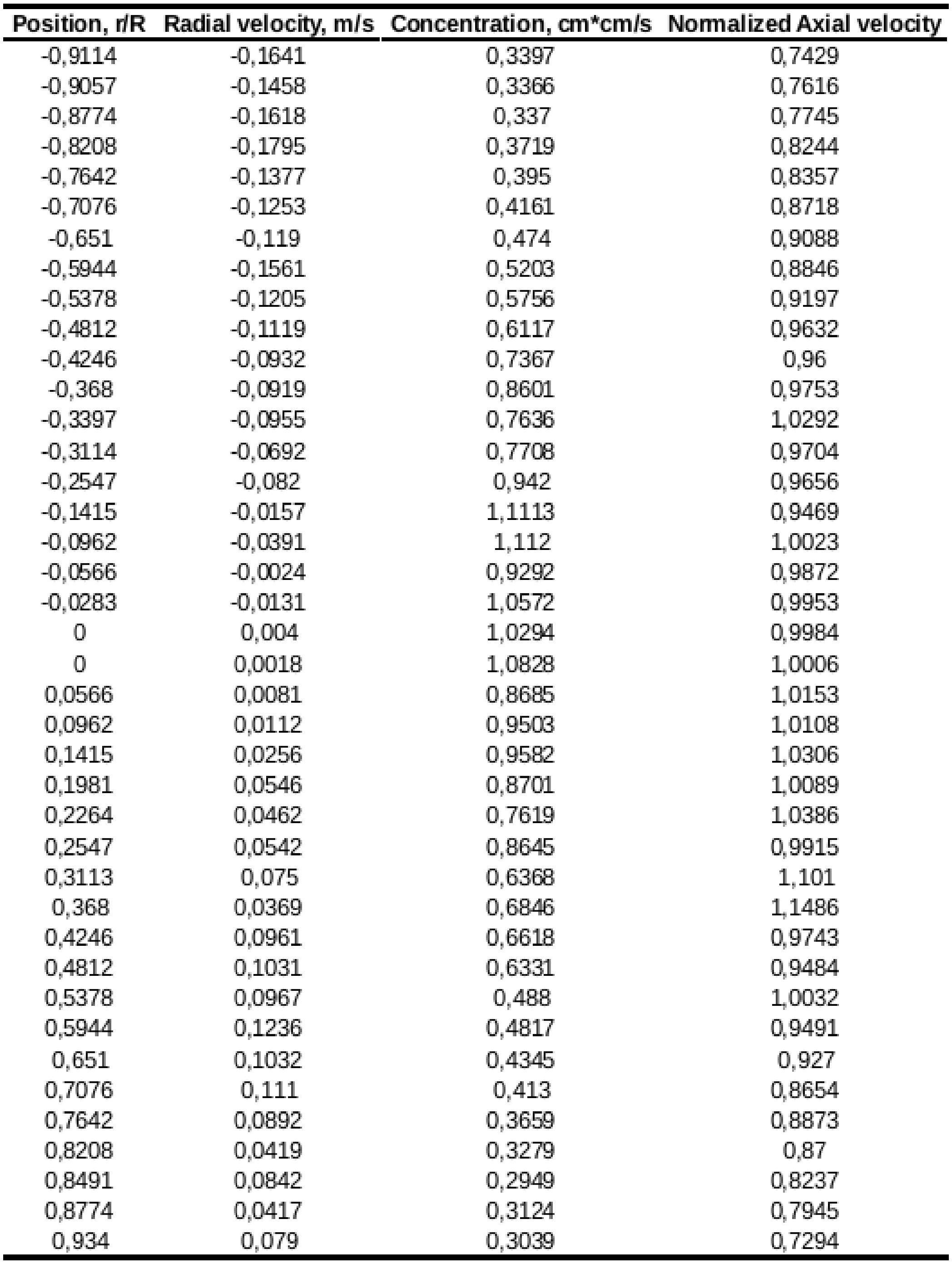
Position, r/R	Radial velocity, m/s	Concentration, cm*cm/s	Normalized Axial velocity
-0,9114	-0,3687	0,0234	0,7915
-0,9057	-0,288	0,0144	0,8175
-0,8774	-0,3745	0,0294	0,7192
-0,8208	-0,4208	0,0374	0,8066
-0,7642	-0,3589	0,0862	0,8626
-0,7076	-0,3081	0,0765	0,8898
-0,651	-0,3112	0,1506	0,9171
-0,5944	-0,2672	0,1642	0,9271
-0,5378	-0,2553	0,2925	0,9566
-0,4812	-0,204	0,3698	0,9324
-0,4246	-0,1778	0,4317	0,9468
-0,368	-0,1811	0,5122	0,983
-0,3397	-0,1729	0,4641	0,9937
-0,3114	-0,1407	0,6649	0,9735
-0,2547	-0,1029	0,729	0,9587
-0,1415	-0,0718	0,9203	0,9678
-0,0962	-0,0331	0,9865	0,9965
-0,0566	-0,0301	1,0402	1,0039
-0,0283	0,0092	1,0827	1,0275
0	0,0067	0,9075	1,0073
0	-0,0159	1,0679	0,9639
0,0566	0,0173	1,0423	1,0067
0,0962	0,0303	0,8124	0,9961
0,1415	0,0481	0,8503	1,0166
0,1981	0,0413	0,7532	0,99
0,2264	0,0869	0,7482	1,0183
0,2547	0,0847	0,5828	1,038
0,3113	0,1359	0,5314	1,0076
0,368	0,1825	0,3914	1,0831
0,4246	0,191	0,2755	1,0124
0,4812	0,196	0,2403	1,0136
0,5378	0,2244	0,1449	1,0015
0,5944	0,2665	0,0914	0,965
0,651	0,3049	0,0525	0,9514
0,7076	0,3099	0,0377	0,9402
0,7642	0,2975	0,0186	0,9251
0,8208	0,2573	0,0123	1,0487
0,8491	0,3517	0,009	0,881
0,8774	0,2797	0,0063	0,8984
0,934	0,1908	0,0048	0,8594

Table VIII.2: Experimental data from ARNASON obtained for particles of $5\mu m$ [4] at $z = 0,502$m.



Position, r/R	Radial velocity, m/s	Concentration, cm*cm/s	Normalized Axial velocity
-0,9114	-0,2765	0,0591	0,7935
-0,9057	-0,2378	0,0516	0,8048
-0,8774	-0,2518	0,0627	0,8006
-0,8208	-0,2656	0,097	0,8482
-0,7642	-0,2613	0,1305	0,8567
-0,7076	-0,2332	0,1761	0,8953
-0,651	-0,2064	0,2303	0,9157
-0,5944	-0,2249	0,2721	0,9387
-0,5378	-0,1996	0,354	0,9486
-0,4812	-0,1685	0,4268	0,941
-0,4246	-0,1513	0,549	1,0051
-0,368	-0,1494	0,6026	1,002
-0,3397	-0,1125	0,7141	1,0023
-0,3114	-0,1258	0,6762	1,006
-0,2547	-0,1025	0,7983	0,9888
-0,1415	-0,0622	1,0046	1,0426
-0,0962	-0,0337	0,9862	1,0067
-0,0566	-0,0119	1,1423	1,0348
-0,0283	-0,005	0,9247	1,0923
0	-0,0034	1,0232	1,0108
0	0,002	1,2082	0,9936
0,0566	0,0153	1,0131	1,1206
0,0962	0,0269	0,9966	1,0321
0,1415	0,055	0,9285	1,0266
0,1981	0,0599	0,7728	1,0363
0,2264	0,068	0,7471	1,0174
0,2547	0,0705	0,6196	1,0758
0,3113	0,1113	0,623	1,0483
0,368	0,1282	0,5501	1,0112
0,4246	0,1514	0,3912	1,0229
0,4812	0,1558	0,2555	1,073
0,5378	0,1742	0,2633	0,9745
0,5944	0,1765	0,2	0,9604
0,651	0,2177	0,1444	0,9866
0,7076	0,2103	0,1102	0,9486
0,7642	0,2135	0,0786	0,9114
0,8491	0,1676	0,0355	0,8642
0,8774	0,1543	0,0403	0,8193
0,934	0,0354	0,0339	0,7956

Table VIII.3: Experimental data from ARNASON obtained for particles of $5\mu m$ [4] at $z = 0,679$m.



Position, r/R	Radial velocity, m/s	Concentration, cm*cm/s	Normalized Axial velocity
-0,9114	-0,1641	0,3397	0,7429
-0,9057	-0,1458	0,3366	0,7616
-0,8774	-0,1618	0,337	0,7745
-0,8208	-0,1795	0,3719	0,8244
-0,7642	-0,1377	0,395	0,8357
-0,7076	-0,1253	0,4161	0,8718
-0,651	-0,119	0,474	0,9088
-0,5944	-0,1561	0,5203	0,8846
-0,5378	-0,1205	0,5756	0,9197
-0,4812	-0,1119	0,6117	0,9632
-0,4246	-0,0932	0,7367	0,96
-0,368	-0,0919	0,8601	0,9753
-0,3397	-0,0955	0,7636	1,0292
-0,3114	-0,0692	0,7708	0,9704
-0,2547	-0,082	0,942	0,9656
-0,1415	-0,0157	1,1113	0,9469
-0,0962	-0,0391	1,112	1,0023
-0,0566	-0,0024	0,9292	0,9872
-0,0283	-0,0131	1,0572	0,9953
0	0,004	1,0294	0,9984
0	0,0018	1,0828	1,0006
0,0566	0,0081	0,8685	1,0153
0,0962	0,0112	0,9503	1,0108
0,1415	0,0256	0,9582	1,0306
0,1981	0,0546	0,8701	1,0089
0,2264	0,0462	0,7619	1,0386
0,2547	0,0542	0,8645	0,9915
0,3113	0,075	0,6368	1,101
0,368	0,0369	0,6846	1,1486
0,4246	0,0961	0,6618	0,9743
0,4812	0,1031	0,6331	0,9484
0,5378	0,0967	0,488	1,0032
0,5944	0,1236	0,4817	0,9491
0,651	0,1032	0,4345	0,927
0,7076	0,111	0,413	0,8654
0,7642	0,0892	0,3659	0,8873
0,8208	0,0419	0,3279	0,87
0,8491	0,0842	0,2949	0,8237
0,8774	0,0417	0,3124	0,7945
0,934	0,079	0,3039	0,7294

Table VIII.4: Experimental data from ARNASON obtained for particles of $5\mu m$ [4] at $z = 1,32$m.