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 2018

Code_Saturne documentation

Code_Saturne version 5.1 tutorial: Particles Dispersion in a Turbulent Pipe Flow

contact: saturne-support@edf.fr

TABLE OF CONTENTS

Part I

Introduction

1 Introduction

1.1 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, possibly integrated in the SALOME platform (then named SALOME _CFD)
- 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* study, for example with the CFDStudy module. For instructions on how to do so, please see [5].

1.2 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:

- presentation of Arnason et al. experimental set-up, flow physics and operating conditions.
- tutorial to create the computational domain (geometry and mesh) using SALOME modules GEOM and SMESH.
- set-up of the single phase flow case in *Code_Saturne* with RANS turbulence modelling.
- set-up of the Lagrangian simulation on a frozen velocity field obtained in the previous section.
- comparison of the results of the numerical simulation with measurements from [4].

Part II

Particles dispersion in a turbulent pipe flow

1 Experimental set-up description and study creation

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.

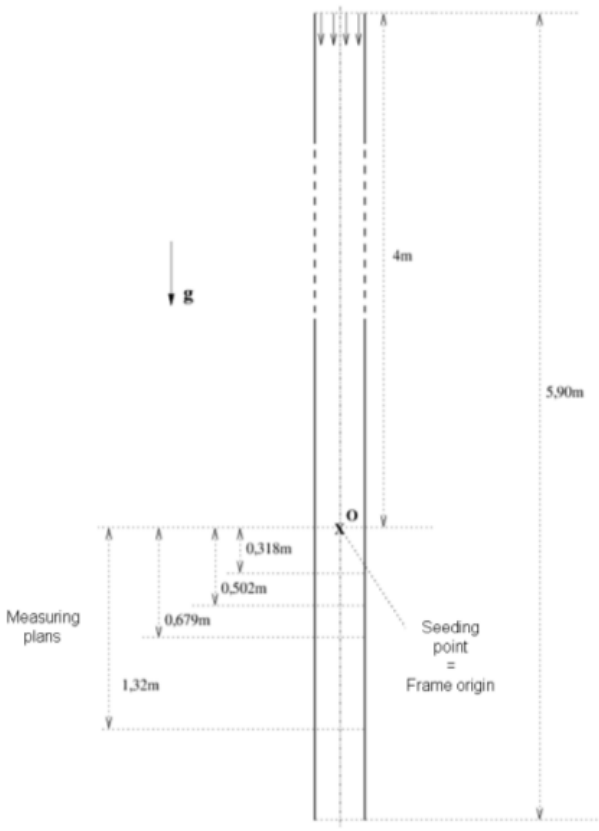


Figure II.1: Schematic description of the Arnason experimental set-up [4]

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 II.3.

$\rho(kg/m^3)$	$\mu(Pa.s)$
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 $d_p^{\mu}(\mu m)$	**Diameter Deviation** $\sigma_p(\mu m)$	**Density** $\rho(kg/m^3)$
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).

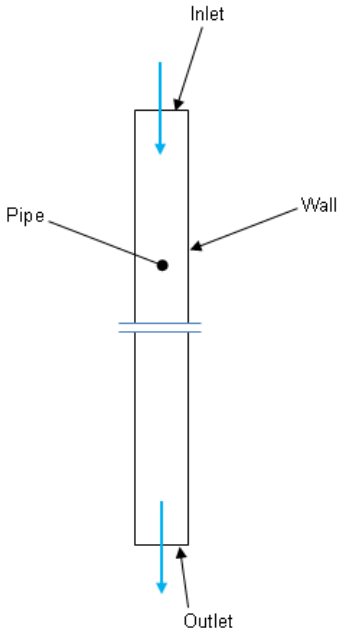


Figure II.2: Location of the boundary conditions.

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.

1.7 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.

1.8 Creating *Code_Saturne* Study

A *Code_Saturne* study called *ARNASON* and a first case are created. This first case will be set up as a single-phase calculation. Call it *RANS_rij_SSG*.

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

- Create the CFD study/case structure with the CFDStudy module
- Save the new file as 'ARNASON'

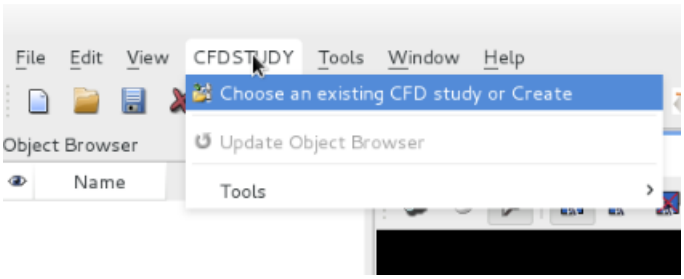


Figure II.3: *Create a CFD study*

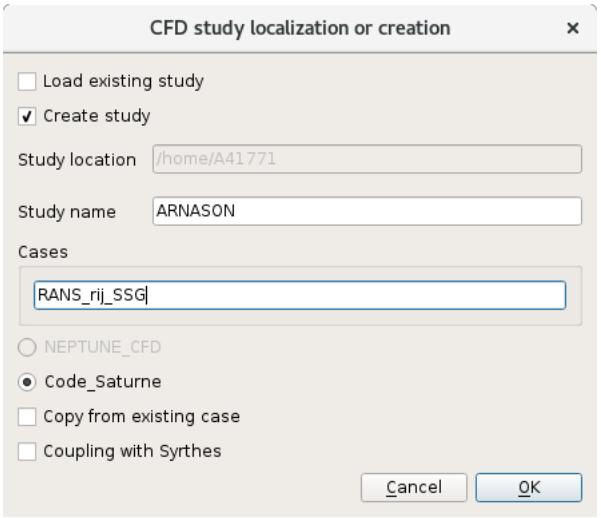


Figure II.4: *CFD study localization or creation*

In the end, you should end up with the directory structure (Figure II.5) shown in the Object Browser tab, displaying the study and the first case.

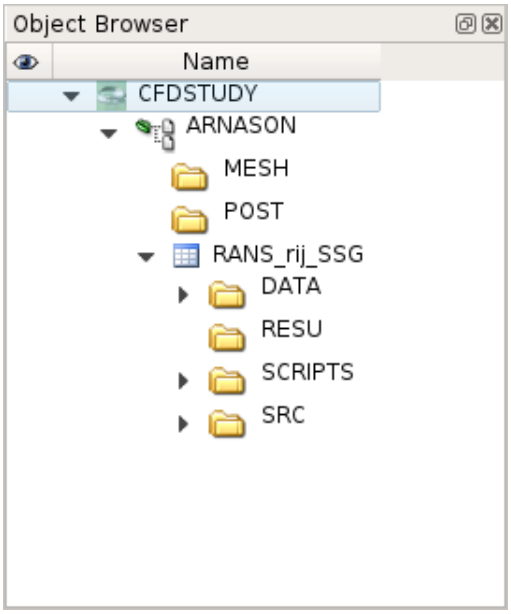


Figure II.5: *ARNASON* Study, *RANS_rij_SSG* case File Structure

The case is then ready to be set up.

2 Creating the Computational Domain

2.1 What you will Learn (Geom and Smesh)

In this part, you will learn how to create the computational domain using the GEOM and SMESH modules of SALOME. Thanks to primitives available in GEOM, creating the CAD will be easy and the work will essentially be in SMESH.

2.2 Creating the CAD

Once the files structure is created by following the step in 1.8, move to the GEOM module in SALOME. We want to 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. In the present part, a disk is created. In the following parts, it will be meshed and finally extruded to generate the two cylinders.

Create a Divided Disk by selecting New Entity » Blocks » Divided Disk. This represents the inlet boundary:

- **name:** inlet_disk
- **Radius:** 0.045
- **Orientation:** OXY
- **Division pattern:** Hexagon

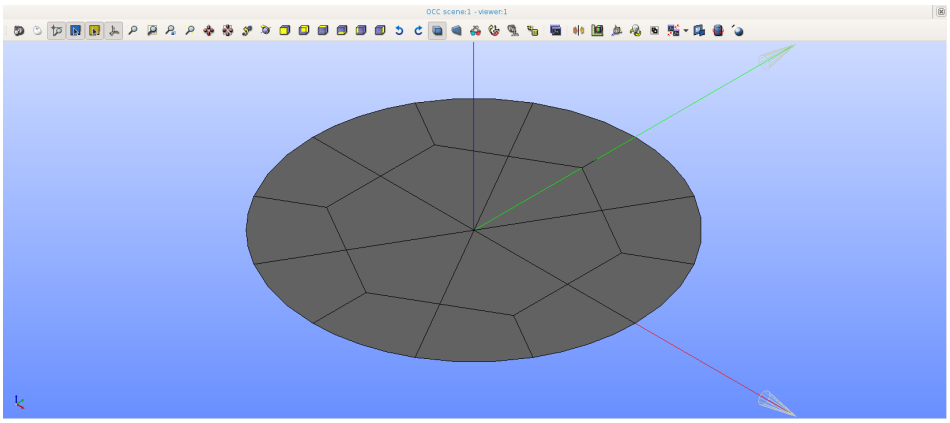


Figure II.6: Divided disk

Finally, create the two following points:

- **name:** z_4.0
- **Coordinates** x: 0; y: 0, z: 4.0
- **name:** z_5.9
- **Coordinates** x: 0; y: 0, z: 5.9

and the two following lines:

- **name:** 0_to_4.0
- **Point 1:** O (origin, clic on O in Object Browser)
- **Point 2:** z_4.0

- **name:** 4.0_to_5.9
- **Point 1:** z_4.0
- **Point 2:** z_5.9

2.3 Creating Groups on GEOM

The next step is to create some groups that will be used as boundary conditions by the numerical model.

Right click on inlet_disk and select **Create Group**. Select the edges composing the border of the disk and add them to the group (Figure II.7). It will be our boundary condition "wall".

- **Shape Type:** edges
- **Group Name:** wall
- **Main Shape:** inlet_disk
- **Main Shape Selection restriction**: No restriction

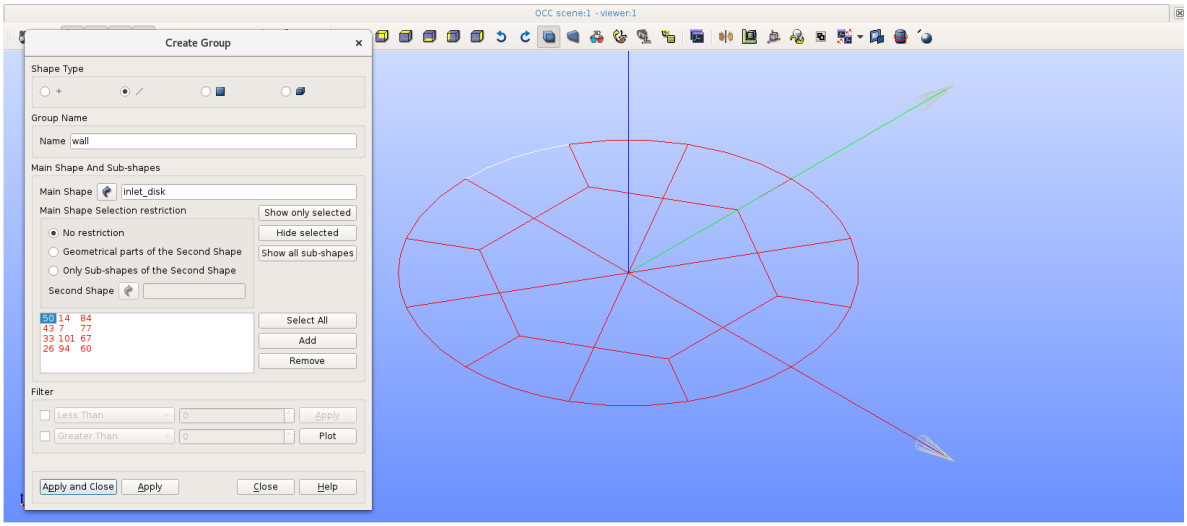


Figure II.7: Creation of the group "wall"

2.4 Meshing

Move to the module 'Mesh' (SMESH) of SALOME. Select inlet_disk in the object browser and Mesh > Create Mesh.

- **Name:** inlet_mesh
- **Geometry:** inlet_disk
- **Mesh type:** Any
- 2D > Algorithm "Quadrangle: Mapping"
- 1D > Algorithm "Wire Discretisation"
- 1D > Hypothesis "Number of segments" with arguments 4 and "Equidistant distribution"

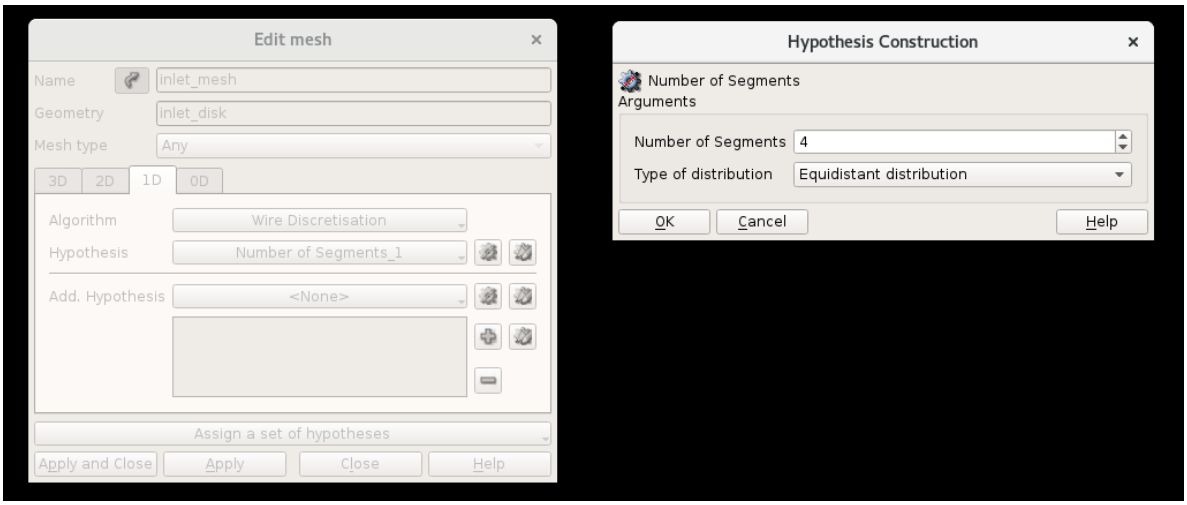


Figure II.8: Create Mesh

Apply.

In order to prepare the extrusion of the inlet disk, the best way is to mesh two lines following the z-axis and extrude the cylinder along these two lines. The refinement of the lines mesh will be the final refinement of the cylinder in the z-direction. Refinement are set as follows:

Mesh of the first section: 0_to_4.0 line

- **Name:** 0_to_4.0_mesh
- **Geometry:** 0_to_4.0
- **Mesh type:** Any
- 1D > Algorithm: "Wire Discretisation"
- 1D > Hypothesis > Number of segments: 100 and "Scale distribution" with a scale factor of 0.045

Apply.

Mesh of the second section:

- **Name:** 4.0_to_5.9_mesh

- **Geometry:** 4.0_to_5.9
- **Mesh type:** Any
- 1D » Algorithm: "Wire Discretisation"
- 1D » Hypothesis » Number of segments 422 and "Equidistant distribution"

Apply.

Then create a mesh group lying on the "wall" group created in GEOM, by right clicking in the object browser on inlet_mesh and selecting Create Groups from Geometry. Add the geometry group "wall" to the Elements list (Figure II.9).

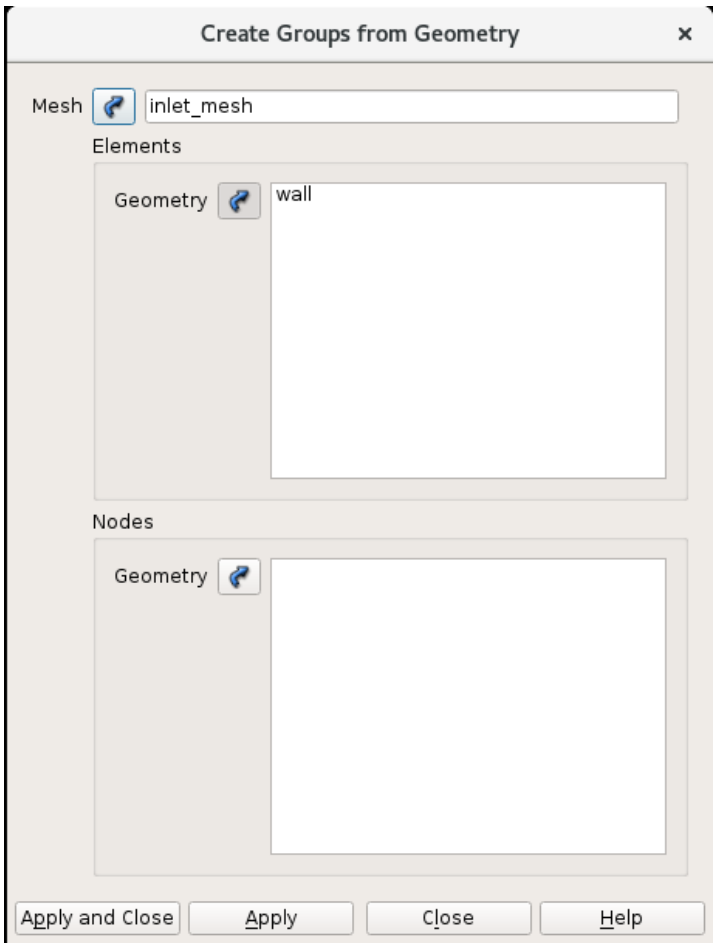


Figure II.9: Create Groups from Geometry

Compute all meshes (inlet_mesh, 0_to_4.0_mesh and 4.0_to_5.9_mesh) by right clicking on each mesh and selecting Compute.

2.5 Creating Groups in SMESH

Right click on inlet_mesh and **Create Group**

- **Mesh:** inlet_mesh
- **Elements Type:** Face

- **Name:** inlet
- **Content:** Select all

Apply and close.

Now the disk is well discretized and just needs to be extruded along each z-section.

For the first section, select Modification > Extrusion along a path and fill the pop-up window as follows:

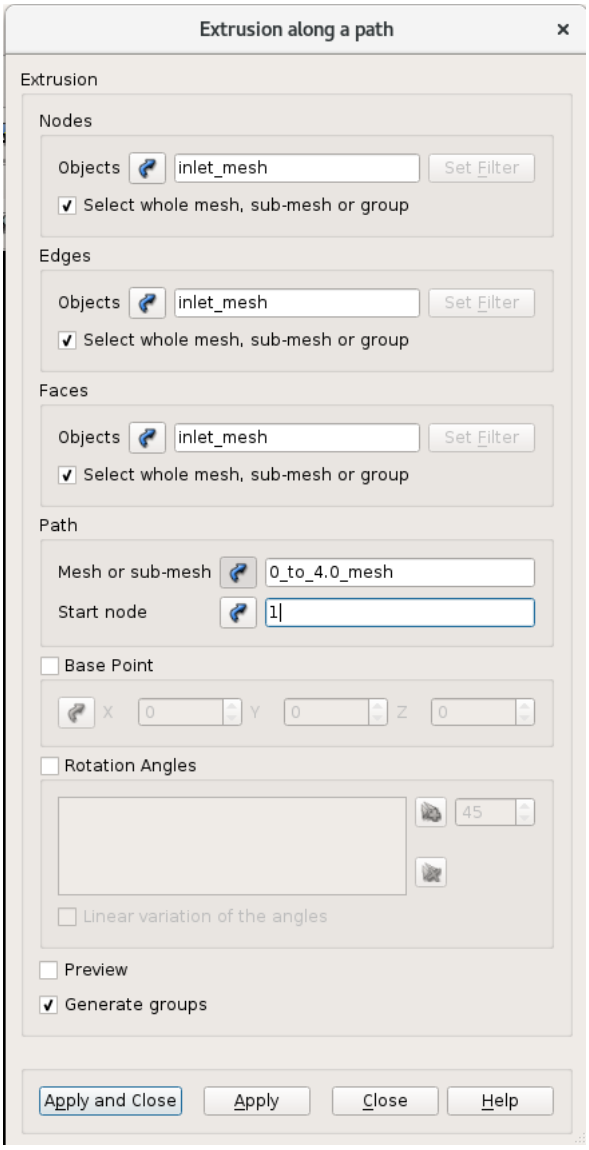


Figure II.10: Extrusion along 0_to_4.0_mesh

Apply and close.

In Groups of Faces, rename inlet_top as injection_plane.

We then need to create a group of cells in which particles will be injected at each iteration of the lagrangian computation. For this, right click on inlet_mesh, and select Create groups. Fill the pop-up window with following settings:

- **Mesh:** inlet_mesh

- **Elements Type:** Volume
- **Name:** injection
- **Group type:** Standalone group
- **Enable manual edition:** toggled

And select manually the cells around the pipe axis at the injection plane as shown on Figure II.11:

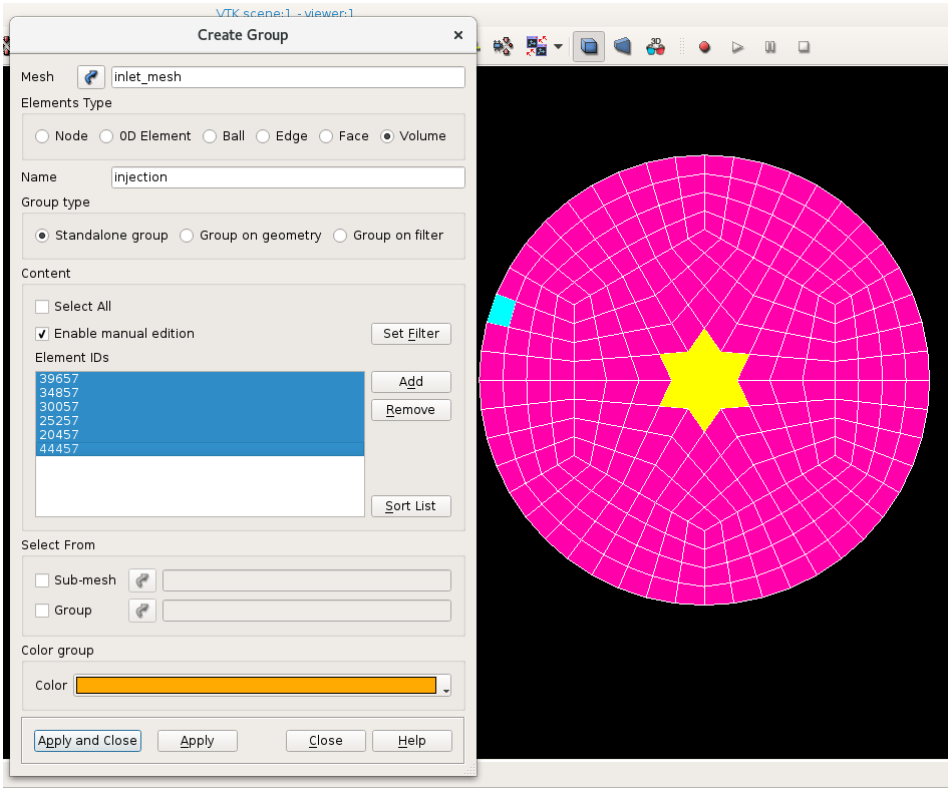


Figure II.11: Creation of the injection group

Finally, select the group injection_plane, created above, and extrude it along 4.0_to_5.9.mesh (Figure II.12):

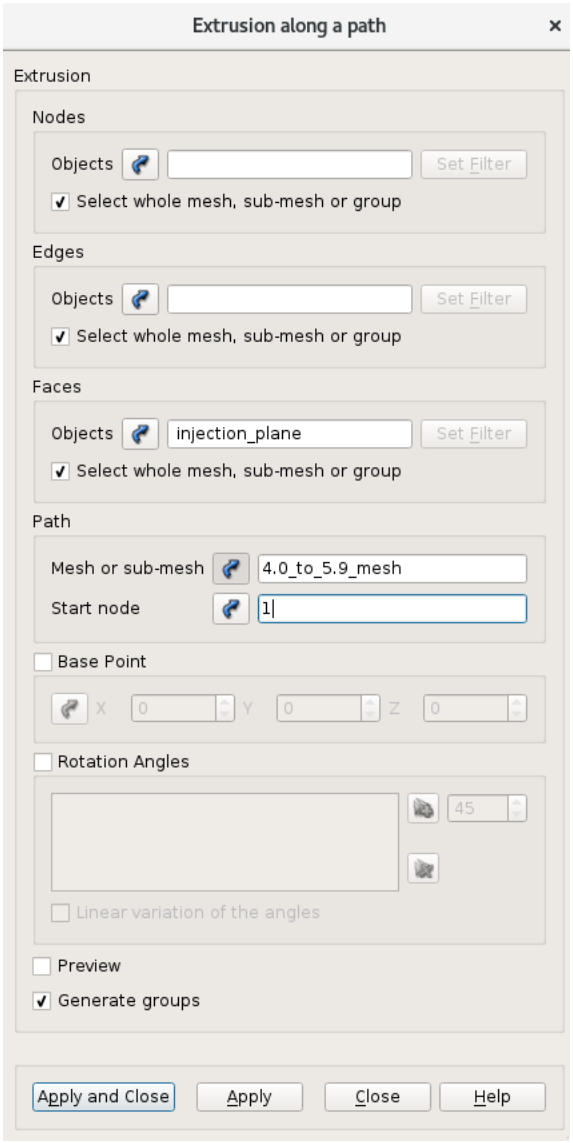


Figure II.12: Extrusion along 4.0_to_5.9_mesh

Remove all groups of Groups of Edges. In Groups of Faces, select wall_extruded and wall_top_extruded (with Ctrl) , click on Mesh > Union of groups and name it wall. Apply and close, then remove wall_top_extruded and wall_extruded. Rename also injection_top as outlet. Finally, remove all groups of Groups of Volumes, except, of course, injection. You can also change the groups colors to be able to distinguish them clearly.

You should get a final mesh as shown on Figure II.13.

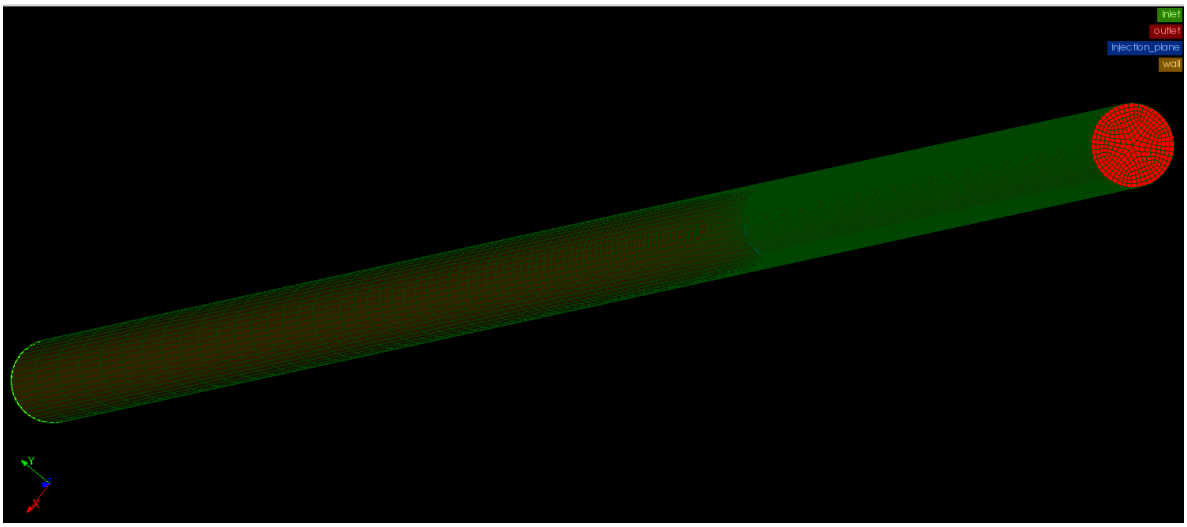


Figure II.13: volume mesh

Two last transformations will be applied to the mesh.

Select Modification > transformation > Symmetry. Toggle Select whole mesh + sub-mesh or group and select inlet_mesh as Name in Arguments. Apply and close.

Select Modification > transformation > Translation and fill the pop-up window as follows (Figure II.14):

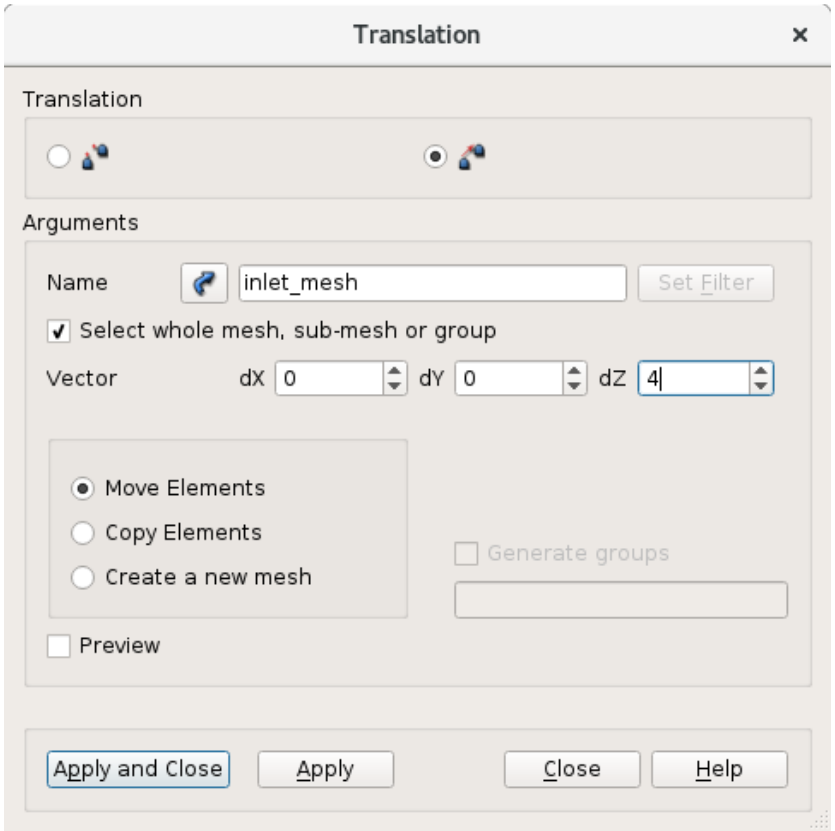


Figure II.14: Translation of mesh

Apply and Close.

The generation of the computational domain is now 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.

3 Single-Phase RANS Computation

3.1 What you Will Learn

In this 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 the previous Section, using the CFDStudy module in SALOME. You will also learn how to integrate user defined functions into a *Code_Saturne* calculation. The user defined functions 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.

3.2 Setting up the CFD Simulation

The CFD case is set-up and run from the CFDStudy module (Section 1.8).

In the CFDStudy module, launch the CFDStudy GUI 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 CFD data file. By default, its name will be setup.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.

Selecting the Volume Mesh

Open the Calculation environment folder and, in the Meshes panel of the 'Meshes Selection', sub-folder, 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 II.15. No further input is necessary for the volume mesh.

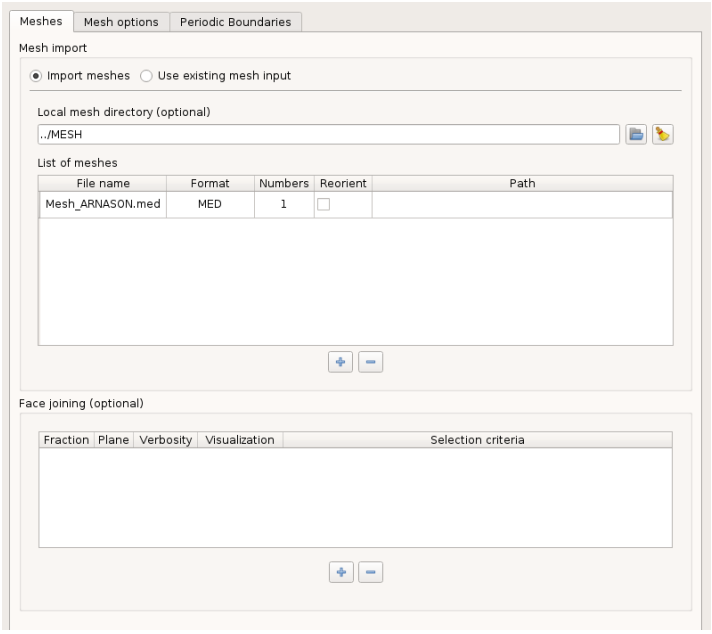


Figure II.15: Importing the mesh.

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

Thermophysical Models

In the **Calculation features** sub-folder, leave all options to their default values, i.e. all specific physics modules should be set to **off** for now.

In the **Turbulence models** sub-folder, change the *Turbulence model* to *Rij-epsilon SSG*. In the **Advanced Options** sub-folder, ensure that the wall function type is set to 'Two scales 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.

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 II.16.

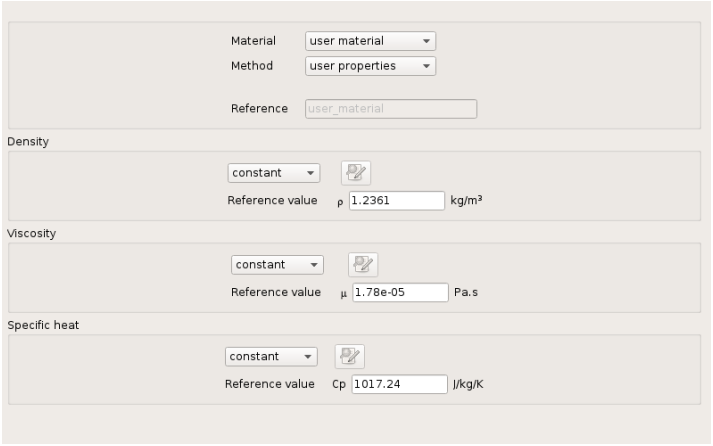


Figure II.16: Specifying the fluid properties.

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.

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.

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 in the 'cs_user_boundary_condition' subroutine so it does not need to be generated here (see 3.2).

First, create the 'wall' and 'outlet' boundary regions by selecting the wall and outlet group in SALOME object browser and by clicking on Add from Salome (Figure II.17). 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'.

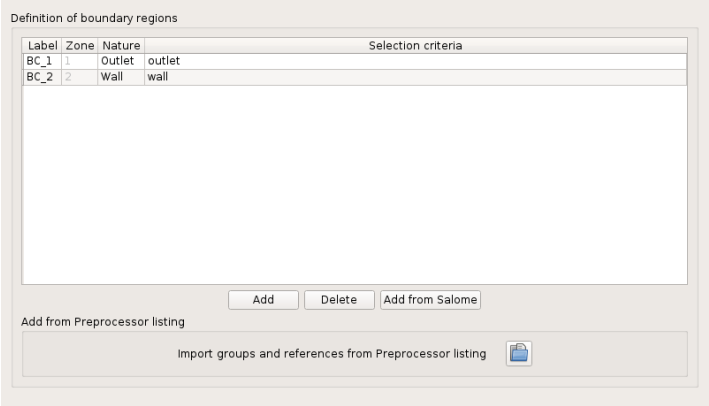


Figure II.17: Defining the boundary conditions.

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.

Numerical Parameters

In the Equation parameters sub-folder, the Solver panel shows that pressure, velocity, turbulent kinetic energy and turbulent kinetic energy dissipation are solved for. In order to decrease overall computation time, it is possible to decrease the solver precision by setting the convergence criterium to 10^{-5} for each variable, except for pressure, without having an impact on solution quality.

In the Time step panel, set the time step to 0.01s and the number of iterations to 850. For this tutorial, this number of iterations is sufficient to reach a steady-state flow solution.

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 recommended 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 II.18. Select the csv format in order to be able to open the output files in ParaVis.

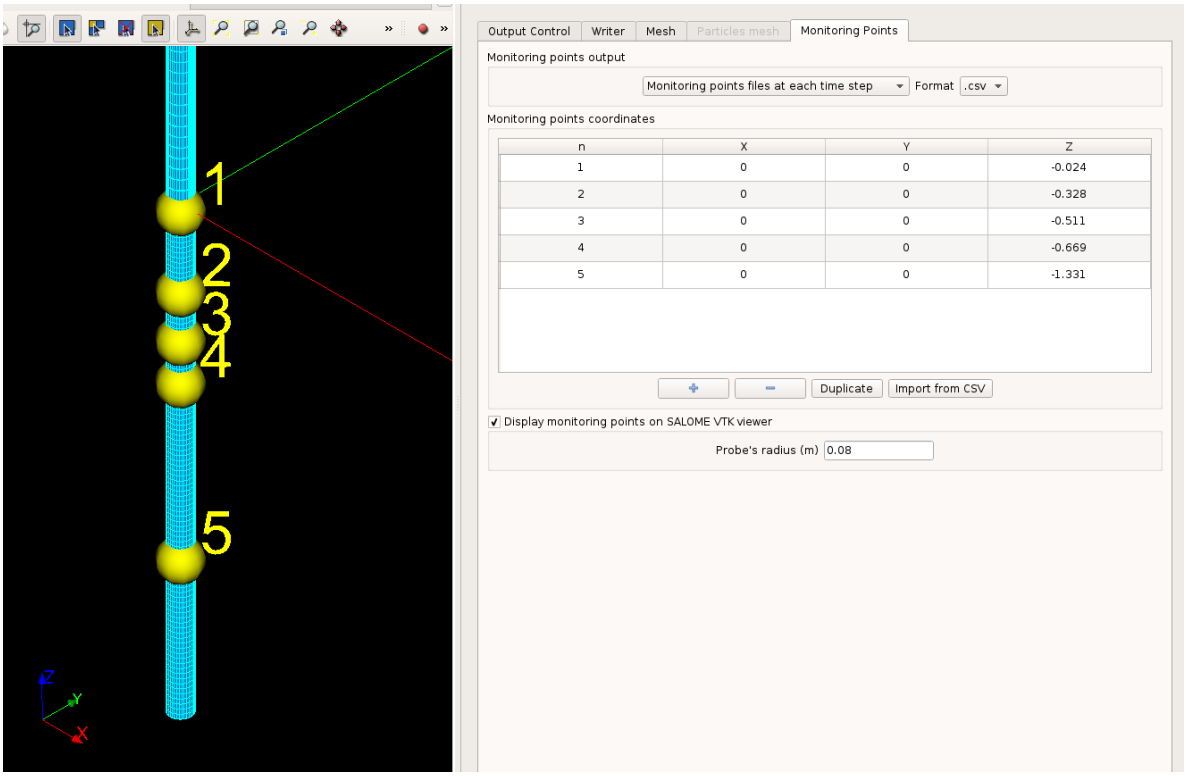


Figure II.18: Monitoring points.

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.

Programming the Inlet Boundary Condition

To begin with, copy the sample file 'cs_user_boundary_conditions-base.f90' from the tutorial /../$ARNASON/RANS\_rij\_SSG/SRC/EXAMPLES$ 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 by a right-click on it in the object browser or by 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 'entre' to all boundary faces

(b) Calculate the V_z velocity component and the turbulence based on the hydraulic diameter

3.3 Running and Analysing the Simulation

Running the Simulation

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

Then, in the panel of Prepare batch calculation sub-folder, all options except the number of processes are let to their default values:

- 'runcase' for the 'Script file'
- '1' for the number of threads per process
- build type to '[default]'.

You may increase the 'Number of processes', depending on the number of cores available on your machine in order to run the simulation in parallel. With the provided mesh (150336 cells), 4 processes would lead to a nearly optimal speed up. To give a rough idea, this calculation can take slightly less than half an hour if run on only one process.

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.

Checking Calculation Convergence

Wait for the calculations to complete and open the 'listing' file in your "ARNASON/RANS_rij_SSG/RESU/DateOfRunTimeOfRun/" directory. Verify that the residuals listed under 'time residual' 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 II.19, II.20).

** INFORMATION ON CONVERGENCE

	Variable	Rhs norm	N_iter	Norm. residual	Drift	Time residual
c	Velocity	0.19438E−03	2	0.94359E−04	0.63501E+04	0.10000E+03
c	Velocity[X]				0.10213E−01	
c	Velocity[Y]				0.10213E−01	
c	Velocity[Z]				0.63501E+04	
c	Pressure	0.38502E−02	42	0.70117E−02	0.99811E+00	0.10000E+03
c	Rij	0.30302E−02	47	0.16926E−01	0.11918E−01	0.98830E+02
c	Rij[XX]				0.39706E−02	
c	Rij[YY]				0.39706E−02	
c	Rij[ZZ]				0.39762E−02	
c	Rij[XY]				0.82075E−65	
c	Rij[YZ]				0.43962E−36	
c	Rij[XZ]				0.42288E−36	
c	epsilon	0.94803E−01	51	0.62111E−01	0.13189E+03	0.10000E+03

Figure II.19: Information on convergence in listing file at first iteration

** INFORMATION ON CONVERGENCE

	Variable	Rhs norm	N_iter	Norm. residual	Drift	Time residual
c	Velocity	0.21440E+00	1	0.11313E−02	0.29963E−03	0.21465E−01
c	Velocity [X]				0.41963E−05	
c	Velocity [Y]				0.41665E−05	
c	Velocity [Z]				0.29126E−03	
c	Pressure	0.13227E−04	13	0.72886E−04	0.23426E−01	0.19810E−02
c	Rij	0.18154E+00	1	0.59522E−05	0.21129E−06	0.80776E−02
c	Rij [XX]				0.10648E−07	
c	Rij [YY]				0.10509E−07	
c	Rij [ZZ]				0.13732E−06	
c	Rij [XY]				0.35421E−08	
c	Rij [YZ]				0.23857E−07	
c	Rij [XZ]				0.25411E−07	
c	epsilon	0.56982E+01	2	0.70640E−05	0.13186E−01	0.26060E−01

Figure II.20: Information on convergence in listing file at first iteration

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. You can use SALOME module ParaVis for this purpose.

0	0	-0.024	0
1	0	-0.328	0
2	0	-0.511	0
3	0	-0.669	0
4	0	-1.331	0

Figure II.21: Monitoring points coordinates

Figure II.22 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 850 iterations. Figure II.23 shows a picture of the magnitude of the velocity at the outlet of the pipe.

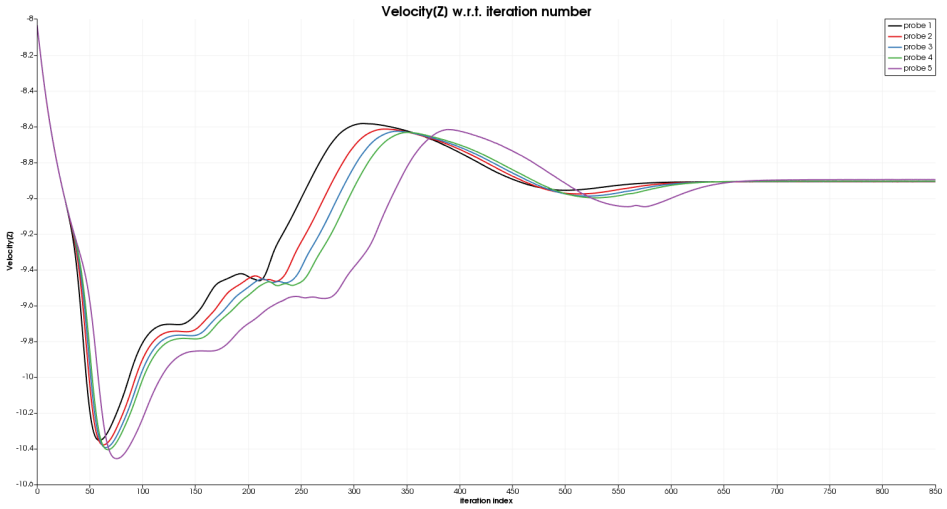


Figure II.22: Evolution of axial velocity during the calculation

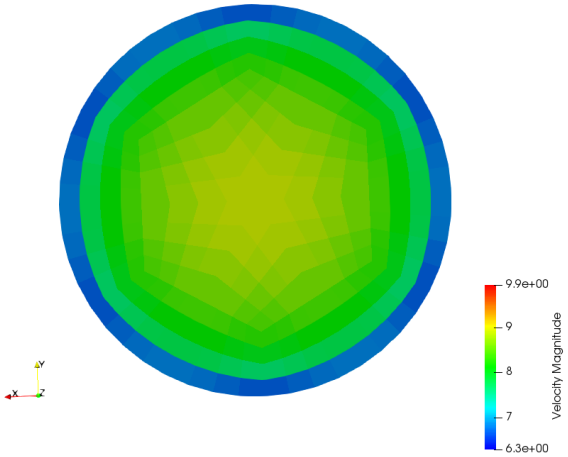


Figure II.23: Velocity magnitude at outlet

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.

4 Two-Phase Lagrangian Computation

4.1 What you Will Learn

In this part of the 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].

4.2 Create a case with "copy-from" feature

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

In order to **avoid setting again** the RANS case as described in the previous section, the second case will be created using the "copy-from" feature. In SALOME object browser, right click on the study name "ARNASON" and select Add case. Then in the pop-up window, enter the name "RANS_rij_SSG_5M" for example, toggle the option copy from existing case and choose the first case directory "RANS_rij_SSG". Finally click on OK.

4.3 Setting up the Lagrangian Simulation

Right click on the file setup.xml in the 'RANS_rij_SSG_5M/DATA' directory which was just created in the object browser and select Open GUI. Check that the directory of the case is 'RANS_rij_SSG_5M and the name of the file is setup.xml (if you see unnamed instead, close the file and repeat the instruction correctly).

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

Setting up the Lagrangian Model

Calculation Features: In the panel of the 'Calculation features' folder, select 'Frozen carrier flow' in the drop down menu of the 'Eulerian-Lagrangian multi-phase treatment' menu as shown in Figure II.24.

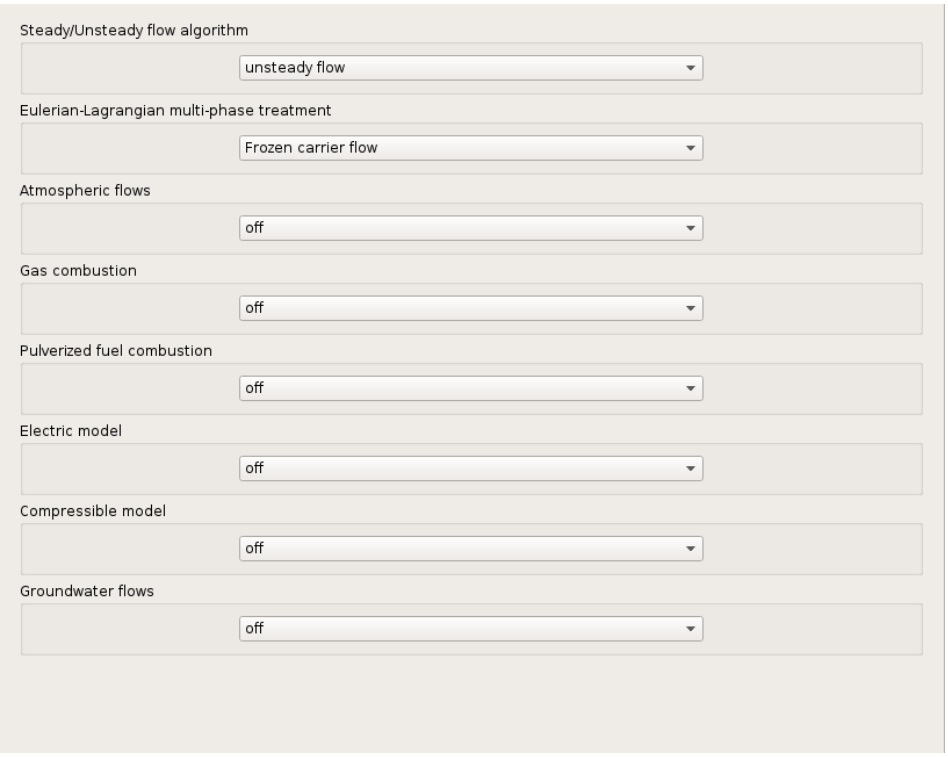


Figure II.24: Selecting the flow physics.

Particles and Droplets 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. By choosing 'Frozen carrier flow' in the previous section, the box 'The continuous phase flow is a steady flow' has been automatically ticked. Leave all other settings in this panel at their default options.

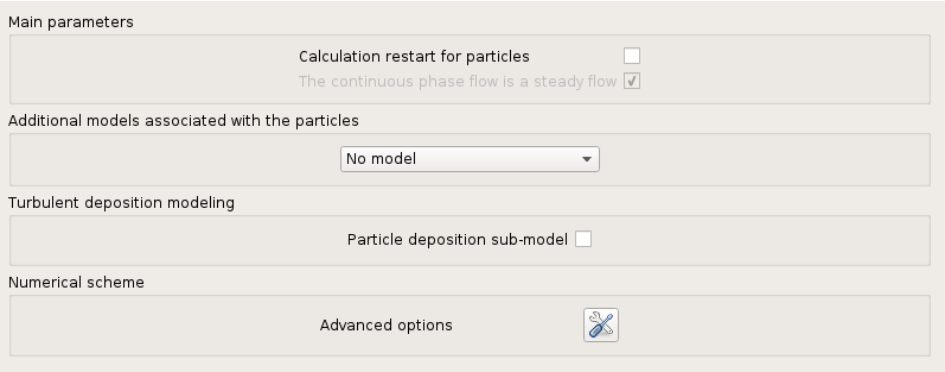


Figure II.25: Global settings menu.

Click on the 'Advanced options' icon to specify the numerical scheme, as shown in Figure II.26. Let the other options at their default values. They should be 'first order scheme' for the 'integration for the stochastic differential equations' option and the 'particle turbulent dispersion' box should be ticked.

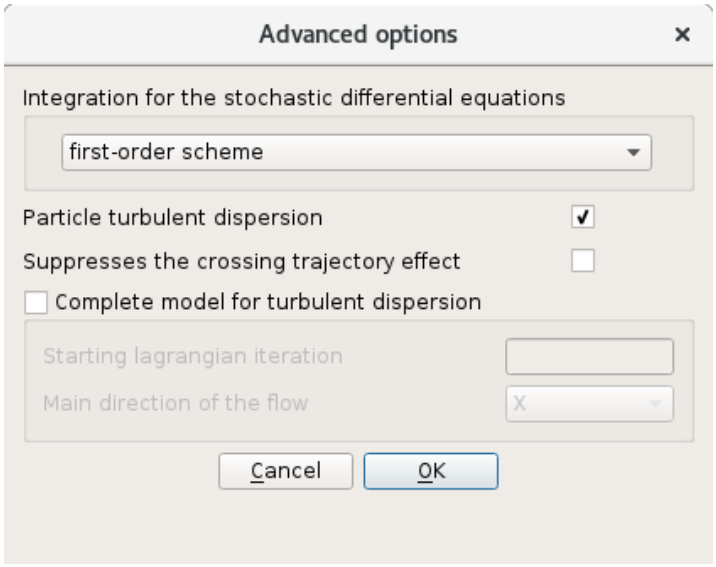


Figure II.26: Advanced options of the numerical scheme menu.

In the 'Statistics' sub-folder, select all parameters, as presented in Figure II.27. The number of particles present in the computational domain is constant after 150 iterations, hence the statistics are started after iteration 400 (cf. Figure II.34 in 5.3).

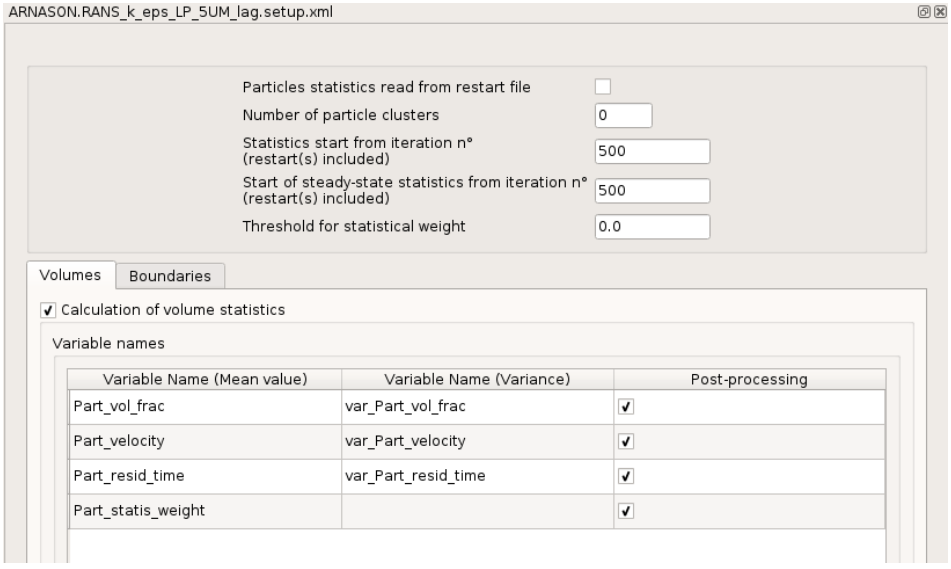


Figure II.27: Statistics menu.

In the panel of the Output control sub-folder, ensure that the 'Output listing at each time step' is set to 1 for "log frequency for particles". This will output particle information at every time step. In the Lagrangian solution control subfolder, select all the options for the different variables to save. The final set-up for this panel is shown in Figure II.28 and Figure II.29.

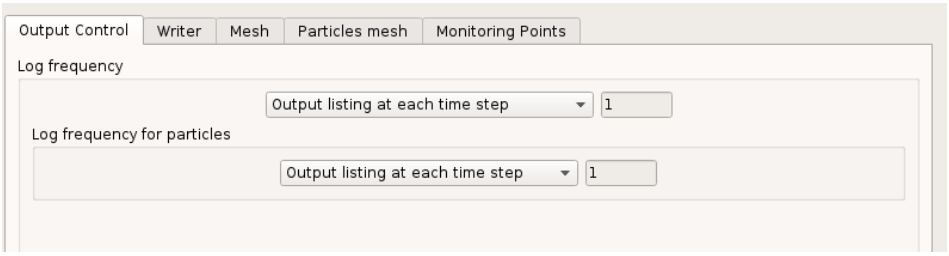


Figure II.28: Output menu.

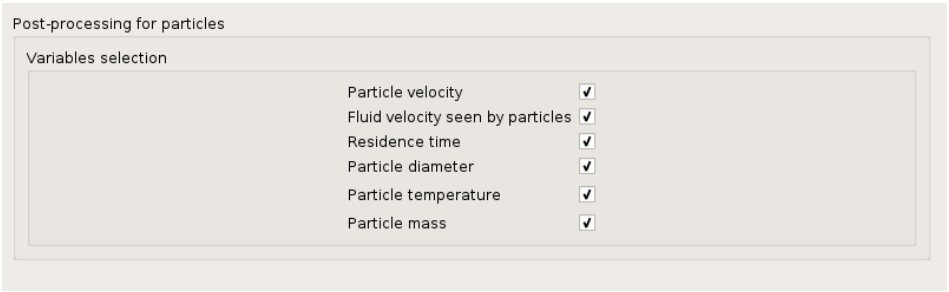


Figure II.29: Solution controls.

Volume conditions In the experiments, the particles are injected further downstream of the pipe's inlet plane. Therefore, we inject particles inside a volume section selected to contain the experiment's injection point. The injection will be defined below using the *cs_user_lagr_injection* and *cs_user_lagr_volume_conditions* user functions but the zone has to be defined in the GUI.

In the sub-folder Volume conditions » Volume region definitions, define a volume zone in which particles will be injected. Name it "particle_injection" and set "injection" in the selection criteria field to define it as shown on Figure II.30. Notice, that this zone has no "nature" defined for now, since its nature will be defined later in *cs_user_lagr_volume_conditions*.

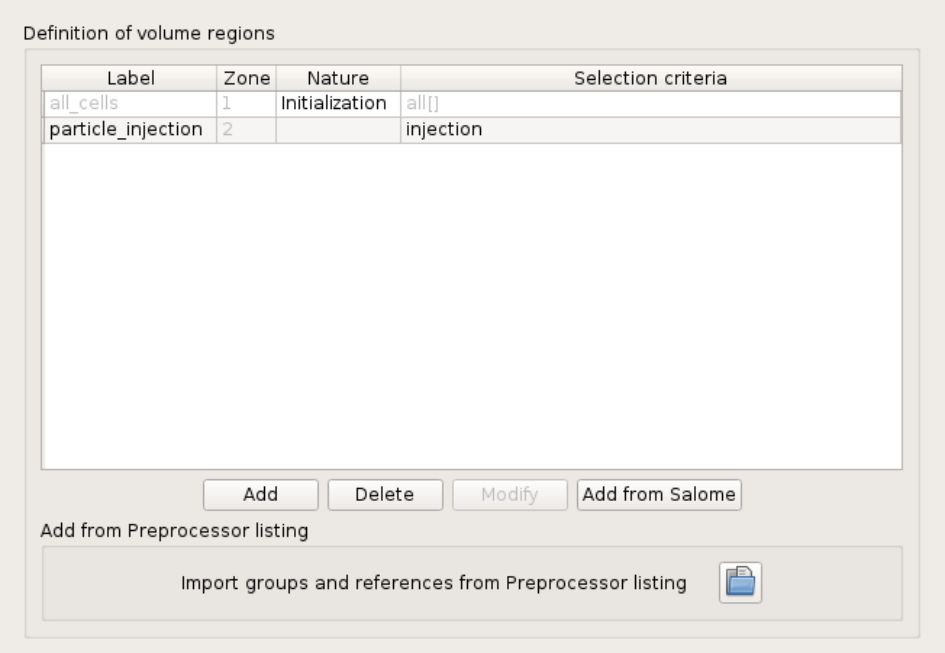


Figure II.30: Definition of the volume zone in which particles will be injected.

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

Add the 'inlet' boundary in the Definition of boundary regions sub-folder, but do not specify the boundaries further in this folder. These will be specified for the particles in the Particles boundary conditions sub-folder.

Go to the Particles boundary conditions sub-folder and ensure that the type of 'Particle-boundary interaction' for the three boundary conditions inlet, outlet and wall are as follows:

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

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.

Numerical Parameters

In the Numerical Parameters folder, leave all settings at the values set for the RANS calculation in the Global parameters and Equation parameters sub-folders.

In the Time step panel 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 II.31.

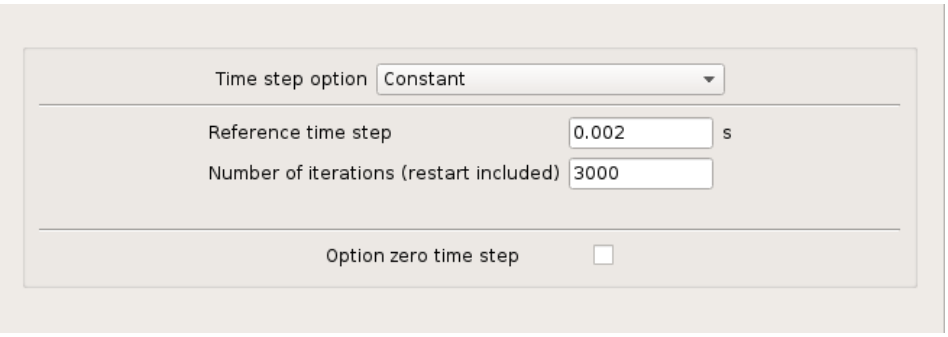


Figure II.31: Time step menu.

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 II.32, so that the particle field is injected on top of the previously calculated single-phase field.

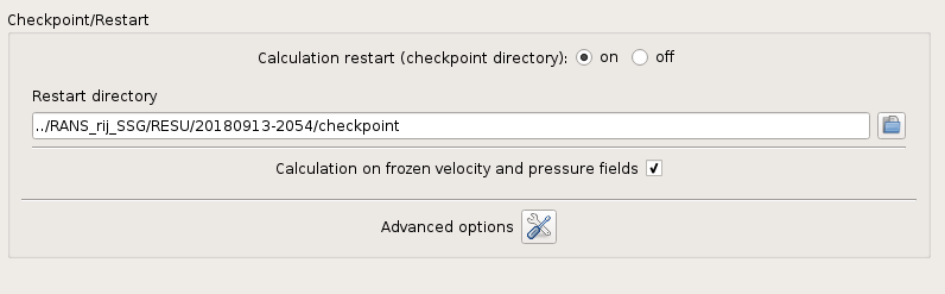


Figure II.32: Start/restart menu.

The set-up of the two-phase flow model in the GUI is now complete. If not already done, you should now save the 'xml' setup file. Before you can run the simulation, user functions in *cs_user_lagr_particle.c*, *cs_user_lagr_volume_conditions.c* and *cs_user_postprocess.c* must be implemented in order to define the injection in the volume and to add output statistics concerning the particles concentrations and the particle axial and radial velocities at the experimental measurement planes.

4.4 Programming in user defined functions

Copy the sample files *cs_user_lagr_particle.c*, *cs_user_lagr_volume_conditions.c* and *cs_user_postprocess.c* from the tutorial directory

/../ARNASON/RANS_rij_SSG_5M/SRC/REFERENCE

to your SRC directory in order to create a local copy. These local copies can be customised and will automatically be compiled and linked to the 'cs_solver' executable at run time.

cs_user_lagr_volume_conditions.c User Functions

Open your local version of the file using the text editor of your choice. The specification of the injection boundary conditions for the particles is done in the *cs_user_lagr_volume_conditions* function. Currently, injecting particles inside the volume is not available using the graphical interface, so programming this function is necessary.

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. Get the volume zone of injection defined in the GUI.
2. Define an injection set for that zone in *cs_user_lagr_volume_conditions*.

cs_user_lagr_particle.c User Functions

Open your local version of the particle tracking C file using the text editor of your choice. The aim here is to set the positions of the injected particles at each iteration on the pipe axis. The customised code is provided with this tutorial which can be used directly or used as a working example.

cs_user_postprocess.c User Functions

The *cs_user_postprocess_writers*, *cs_user_postprocess_probes*, and *cs_user_postprocess_values* functions from the *cs_user_postprocess.c* file are used to generate additional output data relating to the particles. 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. Specify the four measurement planes where the Lagrangian statistics will be calculated
2. Give a name to the four files that will be used by the subroutine to export particle data
3. Initialise the 'mean dispersed phase velocity' and the 'dispersed phase volumetric concentration' parameters which are required to extract particle data (using the graphical interface)
4. 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.

4.5 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 II.33), all options except the number of processes are let to their default values:

- 'runcase' for the 'Script file'
- '1' for the number of threads per process
- build type to '[default]'.

Again, you may increase the 'Number of processes', depending on the number of cores available on your machine in order to run the simulation in parallel.

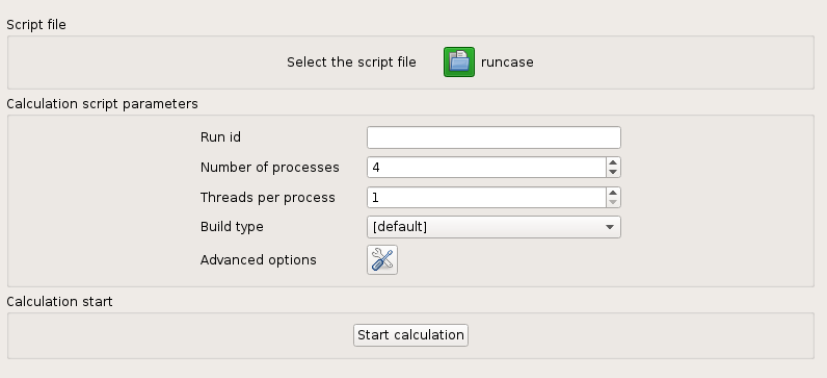


Figure II.33: Batch calculation settings.

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.

4.6 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/RANS_rij_SSG_5M/RESU/DateOfRunTimeOfRun/postprocessing/RESULTS .case'

and to the 'PARTICLES.case' in:

'/../ARNASON/RANS_rij_SSG_5M/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.

5 Results analysis

5.1 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_postprocess.c* user functions.

5.2 Verifying the Simulation

During the Lagrangian calculation, a 'listla' file is automatically created by *Code_Saturne* containing the data listed in Table II.6:

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

Table II.6: Description of the data in the 'listla' file..

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

For example, Figure II.34 and Figure II.35 present, respectively, the number of particles in the domain and the number of particles entering and 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 less than 200 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 analysis.

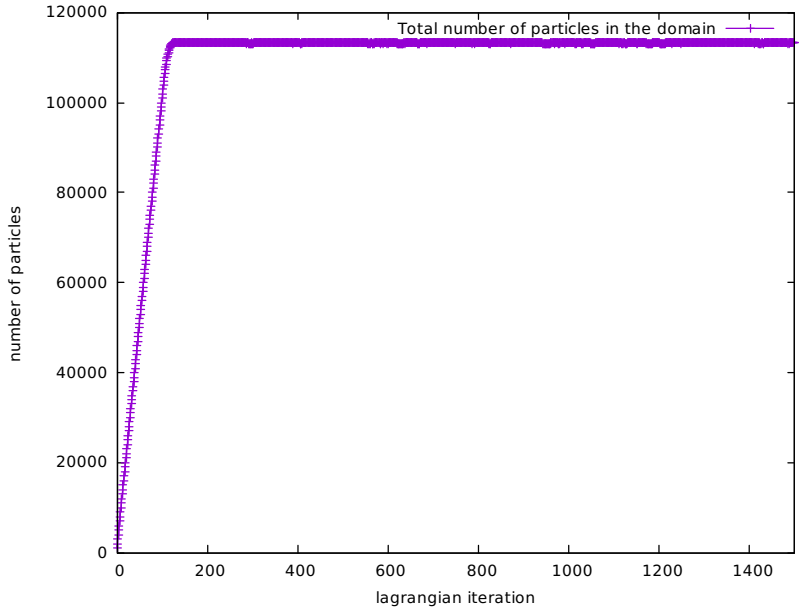


Figure II.34: Number of particles in the computational domain over the first 1500 lagrangian iterations.

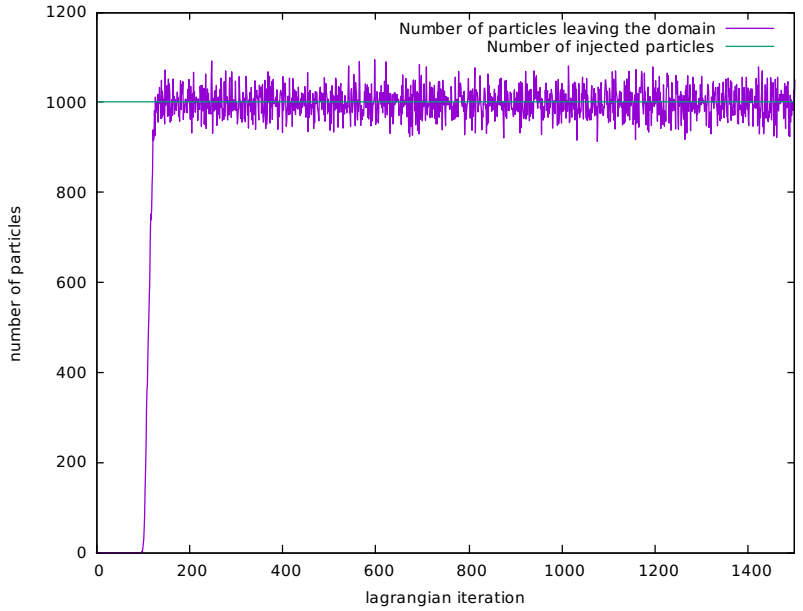


Figure II.35: Number of particles entering and leaving the computational domain over the first 1500 lagrangian iterations.

5.3 Flow Field

Starting with the analysis of the flow field in ParaVis, Figure II.36 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 before spreading in spanwise direction.

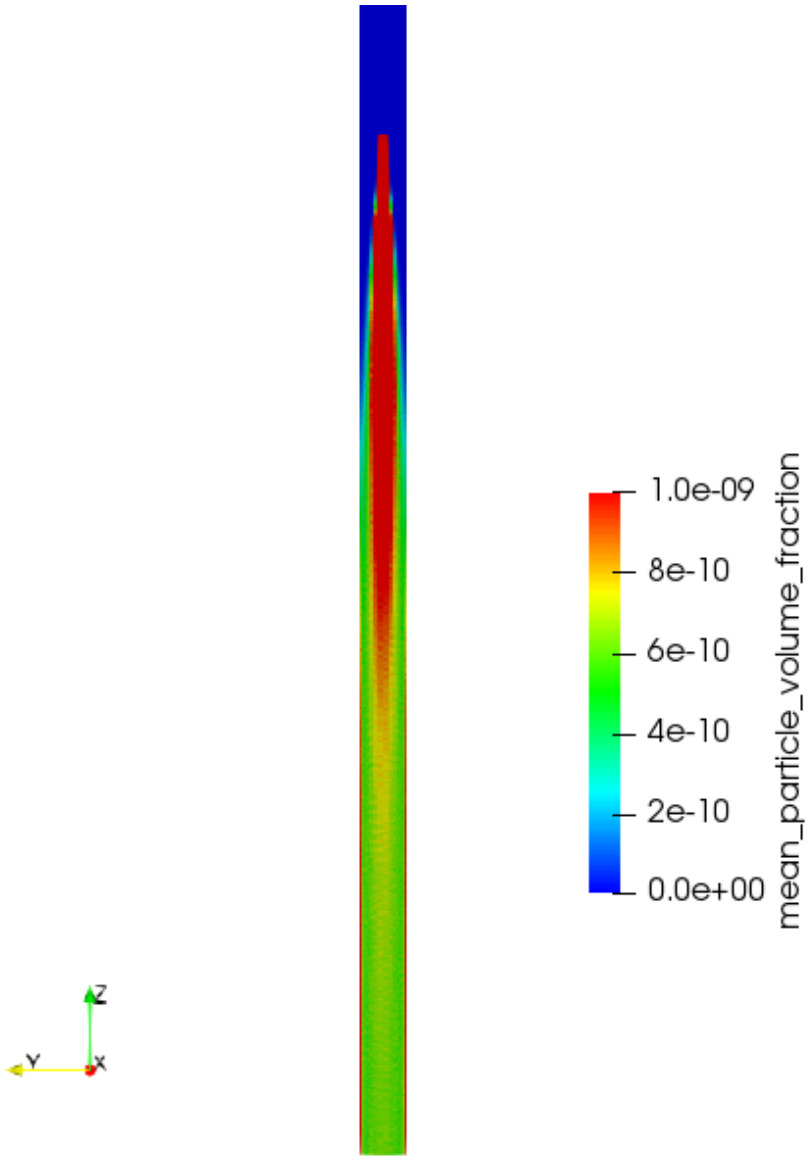
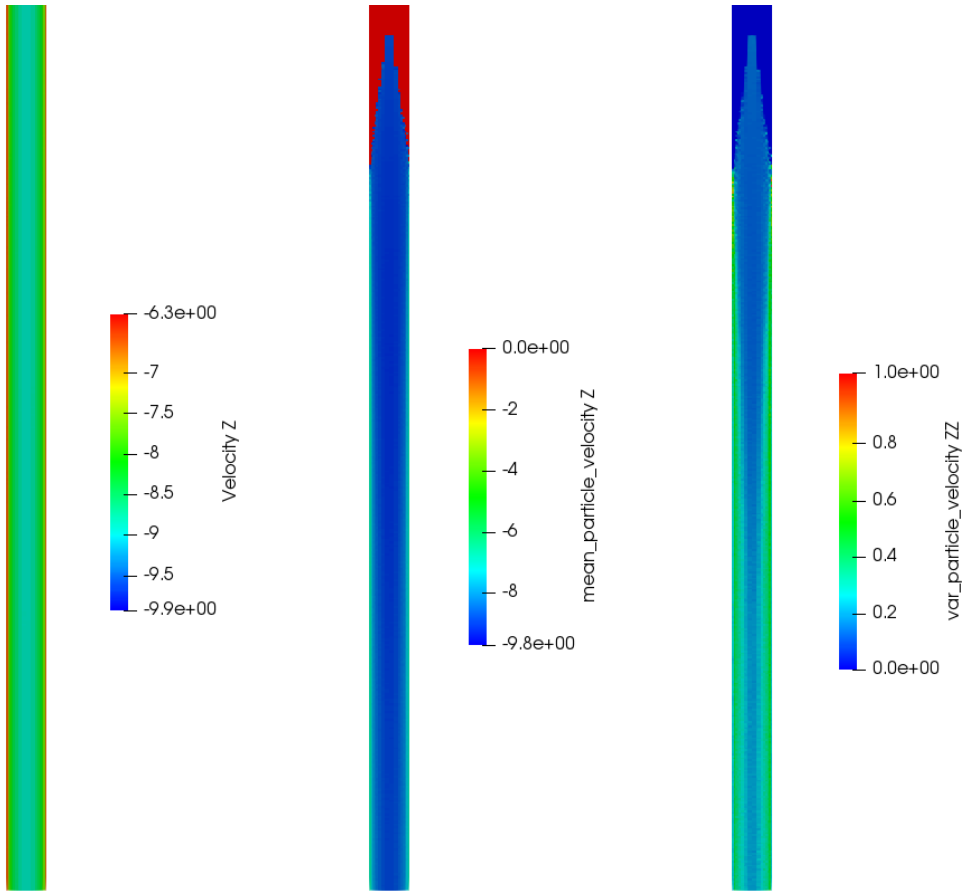


Figure II.36: Volume fraction of the particles in the pipe.

Figure II.37 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 II.37: 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 II.37a and Figure II.37b.

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:

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

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:

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

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

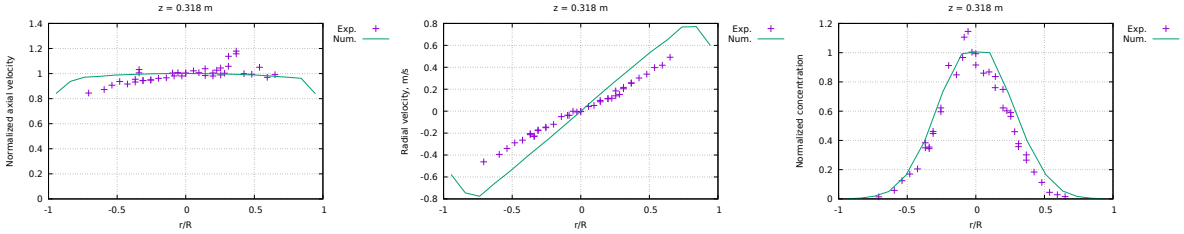
The variance of the vertical velocity of the particles (Figure II.37c) can be seen to be at a minimum along the pipe axis and at its highest close to the flow domain wall, due to the near wall effects such as the boundary layer and the particle rebound condition.

5.4 Comparison of Predicted and Experimental Data

The numerical and experimental data can be compared using the output data files 'Z318.csv', 'Z502.csv', 'Z679.csv', 'Z132.csv' which *Code_Saturne* wrote at the end of the calculation as a result of the programming in *cs_user_postprocess.c* (4.4).

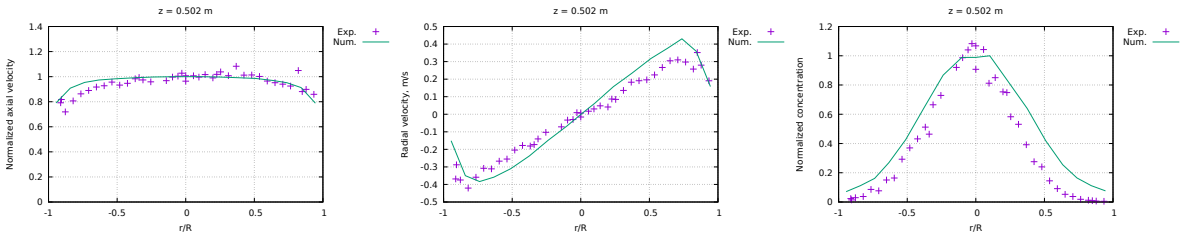
These files contain the particle normalised axial velocity, the particle concentration and the particle radial velocity at the $z = 0,318$, $z = 0,502$, $z = 0,679$ and $z = 1,32m$ planes where experimental data is also available. For convenience, the experimental data at these locations has been reproduced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

Figure II.38 to Figure II.41 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 rather 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.



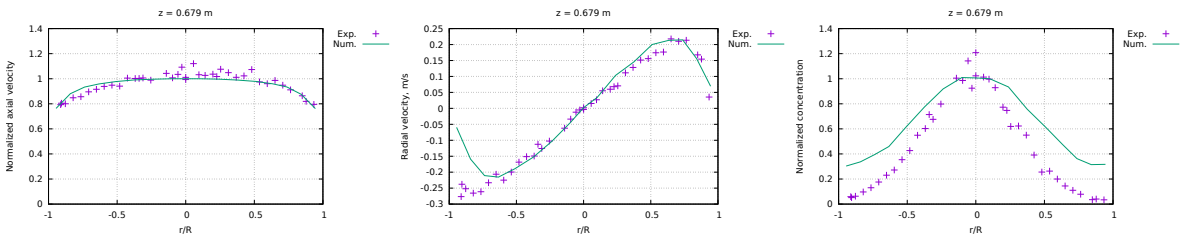
(a) Axial velocity (b) Radial velocity (c) Particle concentration

Figure II.38: Numerical (line) and experimental[4] (symbols) results at z = 0.318m.



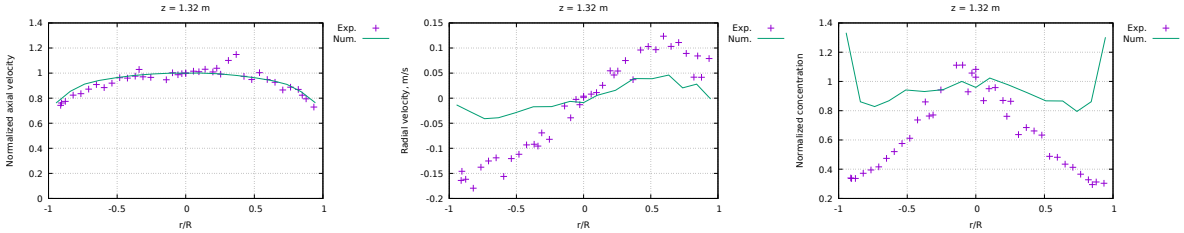
(a) Axial velocity. (b) Radial velocity. (c) Particle concentration

Figure II.39: Numerical (line) and experimental[4] (symbols) results at z = 0.502m.



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

Figure II.40: Numerical (line) and experimental[4] (symbols) results at z = 0.679m.



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

Figure II.41: Numerical (line) and experimental[4] (symbols) results at z = 1.32m.

6 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

1 Appendix A – Experimental Data from [4]

The experimental data from ARNASON [4], for the particles of $5\mu m$ 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 II.7 to II.10 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 II.7: Experimental data from ARNASON obtained for particles of $5\mu m$ [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 II.8: 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 II.9: 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 II.10: Experimental data from ARNASON obtained for particles of $5\mu m$ [4] at $z = 1,32$m.