



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

6, quai Watier F-78401 Chatou Cedex

MARCH 2024

code\_saturne documentation

code\_saturne version 8.0 tutorial: Shear Driven Cavity Flow

 $contact: \ saturne-support@edf.fr$ 



		code_saturne
EDF R&D	code_saturne version 8.0 tutorial: Shear Driven Cavity Flow	documentation Page $1/37$

#### TABLE OF CONTENTS

	I Introduction	3
1	Introduction	4
1.1	TUTORIAL COMPONENTS	4
1.2	TUTORIAL STRUCTURE	4
		_

	II Setting up	5
1	Setting up	6

	III Driven Cavity Flow Case	7
1	General description	8
1.1	What You Will Learn	8
1.2	CASE DESCRIPTION	8
1.3	CREATING THE CODE_SATURNE CASE	9
1.4	Setting up the CFD Simulation	9
2	Simulating within code_saturne	11
2.1	Mesh Tab	12
2.2	CALCULATION FEATURES	14
2.3	Volume conditions	15
2.4	Boundary conditions Tab	17
2.5	TIME SETTINGS TAB	19
2.6	NUMERICAL PARAMETERS TAB	20
2.7	Postprocessing Tab	20
2.8	Running and Analysing the Simulation	24
2.9	Post-processing the Results	25
3	References	37
4	Reference Data from [4]	37

## Part I

# Introduction

## 1 Introduction

#### **1.1 Tutorial Components**

This tutorial makes use of:

- code\_saturne [2], [3] for mesh generation and CFD calculations.
- The SALOME [1] platform for post-processing
- Reference [4] for comparison with published results

The mesh in this tutorial is directly generated using code\_saturne with the *generate cartesian mesh* option. It allows to mesh a cube with a constant mesh size along the three directions. For more complexe geometries and discretisations, it is necessary to use dedicated tools such as the geometry generation and meshing modules of the SALOME platform.

Note that SALOME platform, if it includes the module CFDStudy (code\_saturne integrated inside SALOME), is named SALOME \_CFD. To work through this tutorial you don't need a computer on which SALOME \_CFD is installed or on which you can run a SALOME \_CFD installer. Alternatively, SALOME and code\_saturne can be installed side by side in this order, allowing to specify a salome installation at code\_saturne configuration step.

**Important:** Keep in mind that CFDStudy is just an additionnal module allowing users to keep one window opened, to navigate in and switch modules based on their needs. The aim is to provide an interface including multiple modules.

#### 1.2 Tutorial Structure

This tutorial is made of two sections:

- Section II describes all the procedures required to get along with setting up CFD simulations using SALOME, from code download to case creation.
- Section III illustrates setting up, running, and analysing a CFD simulation entirely with code\_saturne, using the laminar, Shear Driven Cavity as an example. This case is simple but contains enough physics to make it interesting and relevant to practical problems.

If you are already familiar with setting up CFD simulations with SALOME, you may skip Section II and go directly to Section III.

Part II

Setting up

## 1 Setting up

#### **File Structure**

code\_saturne makes use of a pre-set directory structure to access and save input and output files. Simulations are organised in terms of studies and cases. Conceptually, a study contains a series of cases which all rely on the same geometry. The cases represent different instances of simulations for this common geometry, for example for different operating conditions. The pre-set code\_saturne directory structure can be easily created using code\_saturne GUI (Graphic User Interface).

Via the Terminal Make sure to be in the proper directory.

```
$cd 'directory path'
$code_saturne create -s Driven_Cavity -c Reynolds1000
```

The meaning of the pre-set different directories and files is recalled here briefly. For detailed information, please refer to the code\_saturne user manual [3].

- $\boxdot$  DATA directory
  - setup.xml file
    - $\triangleright$  Xml parameter file
  - $\circ$  code\_saturne file

 $\triangleright$  Shell file pointing to the code\_saturne GUI executable

 $\circ \texttt{run.cfg}$  file

 $\triangleright$  Text file with parallel computing parameters, run and slurm options

- ☐ REFERENCE directory
  - > Tabulated thermophysical properties for chemical species and atmospheric properties
  - ▷ cs\_user\_scripts.py file

<sup>L</sup> Python file in which users can modify code\_saturne parameters and settings

•  $\boxdot$  RESU directory

 $\circ$  Initially empty, it will contain the results files and code\_saturne outputs of each code\_saturne run, organised in chronological order

- $\boxdot$  SRC directory
  - $\circ$  User coding source files
  - $\circ \boxdot$  REFERENCE directory
    - $\triangleright$  Templates for all the available user coding files
  - $\circ \boxdot$  EXAMPLES directory
    - $\triangleright$  Practical examples of implementation of user coding for different purposes

Initially, the  $\boxdot$  MESH and  $\boxdot$  POST directories are both empty. By default when run from a case directory, code\_saturne is set up to look for the mesh file in the  $\boxdot$  MESH directory. The  $\boxdot$  POST directory could contain post-processing routines if required.

## Part III

# Driven Cavity Flow Case

## 1 General description

In the second part of the tutorial, the preparation, simulation and analysis of the **Reynolds1000** case of the **DrivenCavity** study is described, from the generation of the cartesian mesh, the data settings, to the running and post-processing of the CFD simulation. All the following steps in this second part are presented using code\_saturne for the CFD simulation, except the post-processing that are done with SALOME.

#### 1.1 What You Will Learn

Through this tutorial, you will learn how to perform an end-to-end CFD simulation using code\_saturne [2, 3] with the SALOME [1] platform, from meshing generation to the computation and post-treatment phases. Specifically, you will:

- Create a simple hexahedral mesh with code\_saturne
- Setup a steady-state, viscous, laminar, isothermal, constant properties fluid CFD simulation with non-slip walls, a moving wall, and symmetry planes with code\_saturne
- Control and run the code\_saturne simulation
- Examine the code\_saturne output and results files, including data along specified profiles and at monitoring points
- Analyse and visualise the results with SALOME ParaVis module

## 1.2 Case Description

A viscous fluid is contained in a box, or cavity. All the walls of the cavity are stationary, but for one wall which is sliding in its plane and sets the fluid in motion inside the box through entrainment. This is a well-know academic case for which published results are available [4].

For the purpose of this example, the cavity is rectangular, two-dimensional, and the top wall of the cavity slides from left to right, as illustrated in Figure III.1 below.



Figure III.1: Schematic of the 2D Lid Driven Cavity.

#### Geometry

The cavity is a square of length L = 1.0 m.

#### **Fluid Properties**

The fluid is given the properties listed in Table III.1 below:

Property	Value	Units
Density $(\rho)$	1.0	$kg/m^3$
Viscosity $(\mu)$	$10^{-3}$	Pa.s

Table III.1: Fluid Properties.

#### **Boundary Conditions**

The problem is considered to be isothermal and the domain is fully enclosed by impermeable, non-slip walls. This means that, exactly at the surface of the walls, the fluid inside the box attaches to the walls and has exactly the same velocity as the walls. Therefore, the problem is fully defined by specifying the velocity of the walls (Table III.2). The side and bottom walls are stationary. The top wall slides in the positive X direction at a speed of  $1.0 \text{ m.s}^{-1}$ .

	Velocity component		
Surface	$V_x \ (m/s)$	$V_y \ (m/s)$	
$W_l$	0.0	0.0	
$W_b$	0.0	0.0	
$W_r$	0.0	0.0	
$W_t$	1.0	0.0	

#### **Flow Regime**

The flow Reynolds number is evaluated to determine whether the flow is in the laminar or turbulent regime. The Reynolds number is calculated based on the lid velocity, the box dimensions, and the fluid properties.

For this case, the fluid properties and lid velocity have been chosen so that  $Re = \frac{\rho \cdot U \cdot L}{\mu} = 1000$ , indicating that the problem is in the laminar flow regime.

#### 1.3 Creating the code\_saturne case

The **DrivenCavity** study and **Reynolds1000** case are created by following the instructions in section II of this tutorial.

The next paragraphs describe how to setup and run the lid driven cavity simulations for a flow Reynolds number of 1000.

#### 1.4 Setting up the CFD Simulation

code\_saturne makes use of a pre-set directory structure to access and save input and output files. Simulations are organised in terms of studies and cases. Conceptually, a study contains a series of cases which all rely on the same geometry. The cases represent different instances of simulations for this common geometry, for example for different operating conditions. The pre-set code\_saturne directory structure can be easily created using code\_saturne GUI (Graphic User Interface)

\$code\_saturne create -s Driven\_Cavity -c Reynolds1000

## 2 Simulating within code\_saturne

Open code\_saturne GUI as explained in Settinp Up chapter of this document and check up your window. The latest should be similar to the figure III.2

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp		
📄 🖆 🖄 🐟 🖻 💀 🧯 💋	) 😰 💶 🌣	
<ul> <li>Calculation environment</li> <li>Calculation features</li> <li>Volume conditions</li> <li>Aut Time settings</li> <li>Aut Time settings</li> <li>Aut Rumerical parameters</li> <li>Postprocessing</li> <li>Performance settings</li> </ul>	Image: Study and case directories         Study       [ShearDrivenCavityFlow/shear_driven_ca         Case       Reynolds1000         Associated case sub-directories         Data       DATA         User sources       SRC         Results       RESU	vity_flow
· · · · · · · · · · · · · · · · · · ·		

Figure III.2: Saving the code\_saturne case.

You can now proceed with setting up the case, in the top down order of the folders in the left hand column of code\_saturne.

Go to the **Mesh** Tab in order to specify the mesh which will be used for these calculations.

EDF R&D

#### 2.1 Mesh Tab

Click on the **Generate cartesian mesh** option and fill information related to number of cells, minimum and maximum x, y and z coordinates as follow (Figure III.3).

	Reynolds1000 : setup.xml - code_saturne	×
<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp		
📄 🗎 🕭 🔶 🔳 🖗	i 🛛 🧕 😰 📲 🌣	
Ø X		
<ul> <li>Calculation environment</li> <li>Mesh</li> <li>Preprocessing</li> <li>Volume zones</li> <li>Boundary zones</li> <li>Calculation features</li> <li>Volume conditions</li> <li>E Boundary conditions</li> <li>Ar Time settings</li> <li>Δx Numerical parameters</li> <li>Mostprocessing</li> <li>Performance settings</li> </ul>	Mesh input Import meshes Ouse existing mesh input Import e Generate cartesian mesh X direction Number of cells 50 xmin 0.0 xmax 1.0 Progression law constant  Y direction Number of cells 50 ymin 0.0 ymax 1.0 Progression law constant  Z direction Number of cells 1 zmin 0.0 zmax 0.01	
	Progression law constant	
	Execution mode	
	Standard Computation	
	<ul> <li>✓ Use unmodified checkpoint mesh in case of restart</li> <li>✓ Save mesh if modified by preprocessing</li> </ul>	

Figure III.3: Selecting the mesh file for the calculations.

Note that boundary zones are automatically generated for each face of the cube. They are named X0, X1, Y0, Y1, Z0 and Z1 where the letter refers to the normal axis and the number to minimum or maximum coordinate.

**Boundary zones** Add three boundary zones by clicking on Add. Fill all fields (referring to III.3) as follows :

Zone	Selection criteria	Nature
1	Y1	Wall
2	X0  or  X1  or  Y0	Wall
3	Z0  or  Z1	Symmetry

Table III.3: Boundary condition table

EDF R&D		c	ode_saturr Shear D	ie v Priv	ersion 8.0 tutorial: en Cavity Flow	code_saturne documentation Page 13/37
			Re	ynolog	1000 : setup.xml - code_saturne	×
<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indo	w <u>H</u> elp					
📄 💼 🏝 🥱 👌			2 6 2 🚺	٩		
	0	×				
🕨 🗟 Calculation enviro	nment	E	Boundary regions de	finitior	I Contraction of the second	
👻 🕅 Mesh			Label	Zone	Selection criteria	
Preprocessing			Sliding Wall	1	Y1	
Volume zones			Chatlana http://	-	V0 V1 V0	
<ul> <li>de Calculation featur</li> </ul>	es		Stationnary_walls		XU OF XI OF YU	
Turbulence mo	dels		Symmetry_Walls	3	Z0 or Z1	
🕞 Thermal model						
Body forces						
📑 Species transp	ort					
Weight Volume conditions	S					
Boundary condition	ons					
Ar Numerical parameters	eters					
► Postprocessing					Add Delete	
Performance sett	ings		Add from preproce	ssor la		
				Im	ort groups and references from preprocessor log	
4	•					

Figure III.4: Boundary Zones creation

No further input is necessary as the faces do not require reorientation, joining, or sub-dividing, and there are no periodic boundaries to be concerned with.

Next, move to **Calculation features** to specify physical fluid properties required for calculation.

EDF R&D

#### 2.2 Calculation features

Leave all the default values unchanged, all choices are disabled (Figure III.5), i.e. the incompressible Navier-Stokes model will be solved.

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp			
📄 🚔 🤌 👌 🖉 🔲 🖗 📔	62 🖬 🖾		
<ul> <li>Calculation environment</li> <li>Calculation environment</li> <li>Mesh</li> <li>Calculation features</li> <li>Turbulence models</li> <li>Thermal model</li> <li>Body forces</li> <li>Species transport</li> <li>Volume conditions</li> <li>Boundary conditions</li> <li>Δt Time settings</li> <li>Δx Numerical parameters</li> <li>Postprocessing</li> </ul>	Flow Models            • Standard Eulerian single phase         • Atmospheric         • Electric arcs         • Groundwater         • Reactive flows (combustion)         • Homogeneous Eulerian - VoF model         • Eulerian multiphase (neptune_cfd)         • Eulerian		
W Performance settings	Additional Features Particles and droplets tracking Turbomachinery model Deformable mesh (ALE method) Fans (source-term model)	off   None	

Figure III.5: Selecting the flow physics.

By default **Deformable mesh (ALE method)** and **Fans (source-term model** are not selected. Continue setting up the **Calculation features**.

**Turbulence models** In **Turbulence models** we need select **No model (i.e. laminar flow)**, which is the correct flow regime for the flow Reynolds number that we have chosen in paragraph 1.2 (Figure III.6).

	Reynolds1000 : setup.xml - code_saturne	×
<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp		
📄 🖻 🌜 🔶 🖻 🖗 📔 💋	o 2 🖸 🔅	
@ X		
Calculation environment	Turbulence model	
🕨 🕅 Mesh		
▼ Øψ Calculation features	No model (i.e. laminar flow)	
Turbulence models		
🚡 Thermal model		
Body forces		
Species transport		
<ul> <li>Wolume conditions</li> </ul>		
Boundary conditions		
▶ ∆t Time settings		
▶ ≥ Postprocessing		
Performance settings		

Figure III.6: Selecting a laminar flow (no turbulence model).

**Thermal model** is inactive by default. As our case is isothermal, it does not need to be visited and changed.

Save the file and, as there are no additional scalars, skip the **Species transport** and **Body forces** (as gravity is not a parameter of this problem (constant density) and **Coriolis source terms** are not taken into account in this tutorial.) categories and move to the **Physical Properties** folder where the thermal and physical characteristics of the fluid of interest for the calculations will be specified.

### 2.3 Volume conditions

Before stepping forward make sure these both options : Initialization and Physical properties are ticked.

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp								
📄 🖆 👌 👌 🖬 🖗 🧧 🖉	Ø	2	<b>※</b>					
Ø								
Calculation environment	Ve	olume zone	types					
Kenter for the fo		7	In this Provide a	Discusional and a section	c	Descrites		Momentum source
Calculation features		Zone label	Initialization	Physical properties	Solid	Porosity	Head losses	term
		all_cells	<b>v</b>	✓				
Boundary conditions								
$\rightarrow \Delta t$ Time settings								
▶ ∆x Numerical parameters								
Postprocessing								
Performance settings								
4								
· /								

Figure III.7: Volume conditions

**Physical properties** First check on **Reference total pressure** value. This should be 101325.0 *Pa* by default. The other default values for velocity and length can also be left unchanged. However, some fluid Properties must be modified by clicking on **Physical properties** tab and, as per Figure III.8 below, altering the values of **Density** and **Viscosity** to the chosen values which are listed in Table III.1 above.

	Reynolds1000 : setup.xml - code_saturne ×
<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp	
📑 🖹 🖄 🔗 🖻 🖗 🧯	2 🙆 😰 🖪 🔅
<ul> <li>Calculation environment</li> <li>Calculation features</li> <li>Wolume conditions</li> <li>Calculation features</li> <li>Calculation features</li> <li>Calculation features</li> <li>Calculations</li> <li>At Time settings</li> <li>Az Numerical parameters</li> <li>Performance settings</li> </ul>	Physical properties     Initialization     Material     Method     weind     Reference total pressure     value   101325.0     Pa     Reference total pressure   value   101325.0     Pa     Reference total pressure   value   101325.0     Pa     Reference temperature   value   293.15   K   (used for properties initialization)        Density   constant   Po   1.0   kg/m³        Viscosity     constant   µ   0.001   Pa.s

Figure III.8: Specifying the Physical properties.

**Initialization** To specify the initialization parameters click on **Initialization** sub-folder. In this sub-folder, press on the icon **P** next to **Velocity** to bring up the **Mathematical Expression Editor** and change the initial first component of the velocity to 0.1 m/s as shown in Figure III.9 below.

User expression	Predefined symbols	Examples	
<pre>velocity[0] = velocity[1] = velocity[2] =</pre>	0.1; 0.; 0.;		
		● <u>C</u> ance	l <u>₹0</u> K

Figure III.9: Specifying velocity initial values.

#### 2.4 Boundary conditions Tab

To define boundary conditions, go to **Boundary zones Tab**. You must define the nature of all boundary zones previously created.

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp		
📄 눹 😒 🔶 🖻 🖗 🚺	o 🛛 🖸 🖉	
Image: Second secon	E     E     E     E     Sliding_Wall     Stationary_Walls     Symmetry_Walls	Nature Wall Wall Symmetry
4 Performance actings		

Figure III.10: Boundary definition

Having defined their type, the exact characteristics of the boundaries must be further specified. Click on the **Boundary conditions** sub\_tabs which presents the list of boundaries which can be further specified (Figure III.12). By default, the wall surface is **smooth** and this parameter does not need to be changed.

**Stationnary wall** Leave all default option as follows

mooth or rough wall		
	● Smooth wall ○ Rough wall	
Sliding wall		

Figure III.11: Specifying stationnary wall.

Sliding wall Click on the Sliding\_wall boundary and activate the Sliding wall selection. Fields then appear for the U, V, and W velocity components of the wall. By default, these velocities are null. Click in the U field and enter 1.0, as shown in Figure III.12.

Smooth or rough wa	u	
	Smooth wall     O Rough wall	
<ul> <li>Sliding wall</li> </ul>		
	U 1.0 m/s V 0.0 m/s W 0.0 m/s	

Figure III.12: Specifying stationnary wall.

**Symmetry planes** The **symmetry boundaries** are fully defined and do not need further specification. Therefore, they do not appear in the list.

The mesh and physics of the problem have now been set up. Now, parameters related to the calculation can be specified.

#### 2.5 Time settings Tab

To set the duration of the run and the time step, move to the **Time settings** folder. Change **Constant** to **Time varying (adaptive)** at the **Time step option**. Then, change the **Stopping criterion** to **Number of time steps** and set it up to 400. Finally, increase the **Maximal CFL number** to 8.0, leaving the other parameters unchanged (Figure III.13). By default, the calculation-restart is disabled in **Start/Restart** and so this sub-tab does not need to be visited.

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp	
📄 💼 🕭 👌 👌 🔳 🖗 💼 💋	) 😰 🔲 🌣
<ul> <li>Calculation environment</li> <li>Mesh</li> <li>Calculation features</li> <li>Calculation features</li> <li>Volume conditions</li> <li>Boundary conditions</li> <li>Start/Restart</li> <li>∆ar Numerical parameters</li> <li>Dostprocessing</li> <li>Performance settings</li> </ul>	Time step option       Time varying (adaptive)         Velocity-Pressure algorithm       SIMPLEC         Reference time step       0.1         Maximal CFL number       8.0         Maximal Fourier number       10.0         Minimal time step factor       0.1         Maximal time step factor       1000.0
	Time step maximal variation 0.1
	Stopping criterion Number of time steps  + 400
٩ ــــــــــــــــــــــــــــــــــــ	

Figure III.13: Selecting the number of iteration and CFL number.

#### 2.6 Numerical parameters Tab

**Equation parameters** In Numerical Parameters, leave the settings unchanged in the Global parameters and move to the Equation parameters sub-folder. In the Scheme tab, apply the Centered scheme and disable the Slope test for the velocity (Figure III.14). The former will provide second-order accuracy in space and the latter is unnecessary for a smooth flow.

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp							
📄 🖹 🔦 👌 🔽 🔽 🔓 💋	ø	😰 🚺 🌣					
0 🕱							
Calculation environment		Solver Scheme	Clipping				
🕨 🕅 Mesh							
▶ Av Calculation features		Name	Scheme	Blending	Slope	Flux	RHS Sweep
<ul> <li>Wolume conditions</li> </ul>			ouncine	Factor	Test	Reconstruction	Reconstruction
Boundary conditions		pressure				V	2
▶ ∆t Time settings		velocity	Centered	1		✓	1
Equation parameters							
▶							
Performance settings							
4							

Figure III.14: Specifying the flux schemes.

### 2.7 Postprocessing Tab

Move to the **Postprocessing** section. **Time averages** are not of interest for these simulations and the sub-folder does not need to be visited. However, we want to keep track of the solution at different monitoring points to see how it evolves during the calculations. Aside from the solution residuals and the minimum and maximum values of the flow variables which code\_saturne outputs during the calculations, tracking the solution at significant monitoring points is a very important mean of gaining confidence in the convergence of the calculations and judging whether calculations have been run for a sufficiently large number of iterations. This is explained further in paragraph 2.8 below.

Click on the **Postprocessing** folder panel. The first three tabs, **Output control**, **Writer**, and **Mesh** are set by default to the correct values for this case and do not need to be changed. The **Log frequency** in the **Output control** tab is set to print the calculations diagnostics such as the residuals at each time step.

Writer In the Writer tab, the format of the results file which will be used for post-processing is already set to EnSight, which is compatible with the SALOME module ParaVis which will be used to post-process the results after the run. The file will be located in the most postprocessing sub-directory of Reynolds1000/RESU/runDateAndTime, where runDateAndTime corresponds to the time at which the run was started. Clicking on the results row brings up the additional information about the Frequency, Time-dependency and Options. As already set by default, the results file will only be written at the end of the run and on the Fixed mesh used for this case. The Options relate to the specific details of the file format. The Mesh tab is already set to output the calculations data in all the fluids cells and at all boundary faces to the results file.

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp					
📄 🖹 🖄 🚖 🔳 🖗 🧯	2 🙆 🕻	2 🚺 🌣			
(					
Calculation environment	Ou	utput Control Writer	Mesh Monitoring Po	ints	
All Mesh     All All All All All All All All All		Name	ld	Format	Directory
Volume conditions		results	-1	EnSight	postprocessing
Boundary conditions				·	
▶ ∆t Time settings					
▶ ∆x Numerical parameters					
💌 🔜 Postprocessing					
Additional user arrays					
Time averages					
Volume solution control				<b>•</b> -	
Surface solution control	Ere	equency			
Profiles		equency			
Derformance settings		No perio	dic output	-	
Terrormance settings		Outpu	t at start of calculation	✓ Output at end of	calculation
	Tir	me-dependency			
			Fixed	mesh 👻	
	Op	otions			
			✓ Separate	sub-writer for each mesh	
			File type	binary (native)	
			Polygons	display 👻	
			Polyhedra	display 👻	
(	•				

Figure III.15: Postprocessing Tab - Writer options

**Monitoring Points** In the **Monitoring Points** tab use the + button in order to add four monitoring points located at (0.25, 0.5, 0.0), (0.5, 0.25, 0.0), (0.5, 0.75, 0.0) and (0.75, 0.5, 0.0) (Figure III.16).

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp					
📄 🖹 🕭 🥱 🕐 🔳 🐼 🧯	2 🙆 🖻 🚺 🌣	¥			
Ø	3				
Galculation environment     Mesh	Output Control	Writer Mesh	Monitoring Points		
Your Calculation reatures     Well Volume conditions     E Boundary conditions     ∆t Time settings	✓ One probe	Monitori per cell 🗌 Activate	ng points files at each tin interpolation	ne step 🔹 Format	CSV 🔻
	Monitoring poin	ts coordinates			
Additional user arrays	n	na	me X	Y	Z
Volume solution control	1		1 0,2	25 0,5	0
Surface solution control	2		2 0,	5 0,25	0
Profiles	3		3 0	5 0.75	0
Balance by zone     Performance settings			4 07	5 0,75	0
			4 0,7		
			• D	uplicate Import from CS	٠V
	Display moni	toring points on SAL	OME VTK viewer		
			Probe's radius (m)		
4					

Figure III.16: Defining the monitoring points.

**Surface solution control** Click on the **Surface solution control** sub-folder and disable **Postprocessing** for **Yplus** and **Stress** as they are not relevant to these simulations (Figure III.17)

	Reynolds1000 : setu	p.xml - code_saturne		×
<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp				
📄 🚔 🕭 🔶 国 🖗 📒	2 🧕 2 🖪 🔅			
0				
Calculation environment	Solution control			
🕨 🕅 Mesh			5.1	
φψ Calculation features	Output label	Internal name	Post-	
<ul> <li>Wolume conditions</li> </ul>	Stragg	strass	processing	_
Boundary conditions	Stress	stress		_
▶ ∆t Time settings	Stress, normal	stress_normal		
<ul> <li>Δx Numerical parameters</li> </ul>	Stress, tangential	stress_tangential		
<ul> <li>Postprocessing</li> </ul>	Yplus	yplus		
Additional user arrays				
Time averages				
Volume solution control				
Surface solution control				
Profiles				
Balance by zone				
Performance settings				

Figure III.17: Selection of output variables on the surfaces of the domain.

**Profiles** Lastly, we want to output one-dimensional profiles of variables along straight lines at the end of the calculations. Click on **Profiles** and add two profiles which go through the centre of the Cavity by clicking on the Add button. The first one for the X velocity along the Y axis and the second one for the Y velocity along the X axis. In turn, specify all the fields listed below the table of profiles, starting with **Filename** and finishing with the variables which are to be stored on output.

For the X velocity profile, choose XVel\_YaxisCenterLine for Filename, and .csv for Format so that the profiles may be read in the ParaVis post-processing module of SALOME. The Output frequency must be set to at the end of the calculation. To define the line, press on the Mathematical expression editor adjacent to Line Definition. The line is defined by the following equation :

$$\begin{cases} x = 0.5 \\ y = s \\ z = 0.0 \end{cases}$$

By definition, s varies between 0 and 1.0. For the **Number of points** enter 50 to account for the 50 cells across the domain's height. Finally, click on **Velocity**[0] and use the  $\rightarrow$  button to add it in the list of variables, and press Add to store the profile in the list (Figure III.18).

Repeat the procedure for the Y-velocity profiles, this time entering **YVel\_XaxisCenterLine** for **Filename**, and the following equation for the line defition :

$$\begin{cases} x = s \\ y = 0.5 \\ z = 0.0 \end{cases}$$

Select the **Velocity**[1] variable instead of **Velocity**[0]. The **Reynolds1000** case CFD simulation is now ready to run.

<u>F</u> ile <u>E</u> dit <u>T</u> ools <u>W</u> indow <u>H</u> elp		
📄 🖆 🕭 👌 👌 🔳 🐼 🧯	2 🚺 🗱	
Calculation environment Calculation features Calculation features Calculation features Calculation features Calculations	Image: Second system         Definition of 1D profiles         Filename         XVel_YaxisCenterLine         YVel_XaxisCenterLine         Filename         XVel_YaxisCenterLine	Variables       Velocity       Velocity       Velocity       Velocity
Balance by zone     Performance settings	Output frequency       at each ''         Line Definition       22         Number of points       50         ✓ One probe per cell       Activa         CourierNb       FourierNb         LocalTime       Pressure         votal_pressure       Velocity	n' time steps • 1 te interpolation Velocity

Figure III.18: Specifying the 1D output profiles.

User expression	Predefined symbols	Examples				
x = 0.5;				User expression	Predefined symbols	Examples
y = s; z = 0.0 ;				x = s; y = 0.5; z = 0;		
		• <u>C</u> ance	ι <u>₹</u> <u>ο</u> κ			● <u>C</u> ancel

Figure III.19: Line Definition - XVel\_YaxisCenterLine(left) and YVel\_XaxisCenterLine(right)

To run the case click on the button

A new window will open. Choose 1 processor for the Number of processes. Leave the **Result** subdirectory name and Threads per process to their default value.

Computation	Advanced			
Script paramet	ers			
		Result subdirectory name		
		Number of processes	1	
		Threads per process	1	
		medds per process	-	
Computation sta	rt			
		Can	cel Apply	Save case and run calculation

Figure III.20: Run

Press the Save case and run calculation button to run the computation.

#### 2.8 Running and Analysing the Simulation

Upon firing the code\_saturne run from the GUI, confirmation that code\_saturne is running, "code\_saturne is running", appears in the window from which the GUI was started. This is followed by further messages indicating what stage the calculation is in, from "Preparing calculation data" to "Post-calculation operations".

Wait for the calculations to complete and enter the **Reynolds1000/RESU** directory or open its contents via a browser to inspect its contents. Explanations of the meaning and purpose of the different output files and directories resulting from a run are available in the code\_saturne Users Guide [3] and are not repeated here. Instead, we highlight individual items which relate to this specific run and how the output information should always be used in order to analyse a calculation.

The  $\bigcirc$  RESU directory now contains a new directory named after the date and time at which the calculation was started, expressed on a 24 hours clock in the format "YearMonthDay-HourMinutes".

In this latter directory, notice in particular:

- The profile files XVel\_YaxisCenterLine and YVel\_XaxisCenterLine, written in .csv format,
- The listing file,
- The monitoring and mostprocessing directories.

With your text editor, open one of the profile files, to inspect its structure. The requested variables listed in column format as a function of the (x, y, z) coordinates of the points along the profile line defined in the GUI.

Open the **listing** file to check that the calculation has converged towards stable values. The minimum and maximum values of the solution variables and the solution residuals for these variables are listed in summary tables at each iteration. By comparing the initial and final values in the **Drift** column for each variable, and their evolution throughout the calculation, check that the residuals have decreased by at least two orders of magnitude.

**\*\* INFORMATION ON CONVERGENCE** 

-	 	 	

	Variable	Rhs norm	N_iter	Norm. residua	l Drift	Time residual
	Velocity Velocity[X] Velocity[Y] Velocity[Z]	0.24719E-03	8	0.14525E-01	0.10121E+00 0.10037E+00 0.84043E-03 0.93180E-29	0.24951E+02
с 	Pressure	0.19612E-03		0.55472E-08	0.10000E+01	0.90909E+01

Figure III.21: code\_saturne Convergence history from the listing file, after 1 iteration.

	Variable	Rhs norm	N_iter	Norm. residual	Drift	Time residual
c c c c c	Velocity Velocity[X] Velocity[Y] Velocity[Z] Pressure	0.28301E-03 0.35145E-10	5	0.92835E-08 0.67751E-07	0.97911E-15 0.37050E-15 0.60861E-15 0.40713E-38 0.67751E-07	0.17206E-06 5 5 7 0.80326E-15

#### **\*\* INFORMATION ON CONVERGENCE**

Figure III.22: code\_saturne Convergence history from the listing file, after 400 iterations.

Inspect one of the monitoring files in the  $\boxdot$  monitoring directory. The value of each variable is recorded in separate files at all the monitoring points defined in the GUI and as a function of iteration.

Finally, verify that the **RESULTS\_FLUID\_DOMAIN.case** file containing all the mesh and output variables information for post-processing has been output in the **mostprocessing** directory.

Having validated the calculation itself, you can now proceed to examining and post-processing the results by returning to the SALOME platform.

#### 2.9 Post-processing the Results

In SALOME, select **ParaViS** from the drop-down module selector in the top menu bar. The name of the module will add itself to the Object Browser list and the ParaView-specific panels and menus will be activated, including a new **ParaView scene viewer** window.

Before loading the run data in ParaView, modify the default colour schemes. To visualise scenes on a white background, which is advantageous for printing, change the default settings for ParaView by clicking on File Preferences in the top menu. For ParaViS, in the ParaView Settings tab, change the colours to black for the foreground and text, white for the background, and grey for the surfaces, as shown in Figure III.23 below. Press Apply to enforce the new settings and OK to validate your selection when you are satisfied with the changes.



Figure III.23: Setting the colour preferences for ParaView.

The data to post-process can now be imported in ParaView.

First, you are going to load the monitoring point data to validate that a stable, steady-state solution has been obtained. From the top menu bar, select File Open ParaView File. In the Open File pop panel, navigate to the monitoring sub-directory and select three files. The multiple selection is performed by holding the Ctrl key down as you select the files. Select probes\_Pressure.csv, probes\_Velocity[X].csv, and probes\_Velocity[Y].csv. Close the panel by clicking OK. The three sets of data are now displayed under their file name in the Pipeline Browser. For each file, press Apply to load the data. By default, the data is visualised in tabular form in the ParaView scene viewer. Close the View by clicking the cross button at the top, right hand corner of the view. Then, click on the D button to replace the data view by a Create View menu (Figure III.24).

Create View
Render View
Render View (Comparative)
Bag Chart View
Bag Plot Matrix View
Bar Chart View
Bar Chart View (Comparative)
Box Chart View
Eye Dome Lighting
FunctionalBag Chart View
Histogram View
Line Chart View
Line Chart View (Comparative)
Line Chart View Columns
Orthographic Slice View
Panoramic Projection View
Parallel Coordinates View
Plot Matrix View
Point Chart View
Python View
Quartile Chart View
Slice View
SpreadSheet View

Figure III.24: ParaViewCreate View menu.

Click on the Line Chart View button and prepare the line plots, starting with the pressure data. Make the data visible for that file by clicking on the eye symbol  $\odot$  next to the file name in the **Pipeline Browser**. Go to the **Display** tab and set **Attribute Type** to **Row Data**. In the **Series Parameters** category, unselect the **time** variable. The variables **1**, **2**, **3**, and **4**, representing the pressure at each iteration at the four monitoring points you specified should be selected and the graphs of their evolution as a function of the number of iterations (time steps) should now be displayed in the view window. For clarity, change the legend for each variable. In the **Series Parameters** category, left-click on the row for variable **1** to select it. Double left-click on the name of that variable, in the **Legend Name** column to edit it. Change the name of each variable to **Pressure (1)**, **Pressure (2)**, **Pressure (3)** and **Pressure (4)** (Figure III.25 and III.26).

roperties			Ø
<b>€</b> <sup>#</sup> Apply	🖉 Reset	💢 Delete	?
Search (use l	Esc to clear text)		503
🐈 Propertie	s (probes_Pressure	.csv)	D C 4
💻 Display ()	(YChartRepresenta	tion)	🗅 🗈 C 🛓
Attribute Type	Row Data		*
X Axis Parame	ters		
✔ Use Index Fo	r XAxis		
X Array Name			~
Series Parame	ters		
Variable		Legend Name	
✓ 1	Pressure (1)		
✔ 2	Pressure (2)		
✔ 3	Pressure (3)		
✔ 4	Pressure (4)		
t	t		
Line Thickness		2	\$
Line Style		Solid	•

Figure III.25: Specifying the line plot's legend.

Properties	(	0 1
€ <sup>#</sup> <u>A</u> pply	Image: Constraint of the section o	
Search (use E	sc to clear text)	202 202
📟 View (Line C	Chart View)	
Title		
Chart Title	Use \${TIME} to display current time	
Annotation		
🗹 Show Legend	1	
Left Axis		
Left Axis Title	Pressure (Pa)	
Left Axis Range		
🗆 Left Axis Log	Scale	1
🗆 Left Axis Use	Custom Range	1
Bottom Axis		
Bottom Axis Title	lteration	
Bottom Axis Ran	nge	
Bottom Axis L	Log Scale	1
🗆 Bottom Axis l	Use Custom Range	U

Figure III.26: Specifying the line plot's axes.

In the View window's tool bar Change the **Bottom Axis** title to **Iteration**, and the **Left Axis** title to **Pressure (Pa)** (Figure III.27). The graph shows that the pressure at the four monitoring points becomes constant after about 100 iterations (Figure III.27).



Figure III.27: Pressure at the four monitoring points as a function of iteration.

Repeat the same operation for the remaining velocity files. For both files, change the left axis title to **Velocity** ( $\mathbf{m/s}$ ). For the X velocity, change the variable names to **X Vel** ( $\mathbf{i}$ ), where i = 1, 2, 3, and 4. Keep the default colours for each line so that they are consistent for each monitoring point. For the Y velocity, change the variable names to **Y Vel** ( $\mathbf{i}$ ), where i = 1, 2, 3, and 4. The graphs also confirm that the velocity components at the four monitoring points settle to a stable, constant value after about 100 iterations, indicating that the calculations are converged (Figure III.28).



Figure III.28: Velocity components (X (left), Y (right)) at the four monitoring points as a function of iteration.

You may now post-process the results from the calculations. From the top menu by selecting File Open ParaView File, or by right-clicking on the builtin object in the Pipeline Browser panel and selecting Open, access the pop-up Open File panel and choose the RESULTS\_FLUID\_DOMAIN.case file from your run. The RESULTS\_FLUID\_DOMAIN.case object will be added to the Pipeline Browser and its contents displayed in the Object Inspector panel underneath. Press [Apply] to load the data in ParaViS.

Sometimes you need to extract the computed data in terms of fluid domain and boundary data, to do so, go to the top menu bar and click on Filters Alphabetical Extract Block. The new object ExtractBlock1 will appear in the Pipeline Browser. With the object highlighted, move to the Block Indices panel of the Properties tab. In the data tree, under Root, select Fluid domain to visualise the data calculated at the cell centres inside the cavity walls. Here, we already have two separated cases (Fluid domain and Boundary) so we will only have to select RESULTS\_FLUID\_DOMAIN.case and press (Apply), as shown in Figure III.29.

Pipeline Browser	Ø
builtin:	
RESULTS_FLUID_DOMAIN.case	
Properties	Ø
Apply @Reset	X Delete ?

Figure III.29: Extract the Fluid domain block in ParaViS.

Left click on the **Display** tab of the **Object Inspector**. In the category **Coloring**, two drop-down menus allow you to choose the variable to visualise. Choose **Velocity** in the first one and **Magnitude** in the second one, as shown in Figure III.30.

Coloring			
礡 Velocity	~	Magnitude	~
🞴 Edit		) 🛱 🔯 📘	e



Contour plots of the velocity magnitude are then displayed in the **ParaView scene viewer**. By default, the colour scale is set to the **Cool to Warm** (RGB) colour scheme which is inadequate.

Coloring			^
礡 Velocity	<b>~</b>	Magnitude	
🞴 Edit			
Styling		Choose preset	

Figure III.31: Color map editor.

To modify the visualisation colour scheme, click on the **Choose Preset** button (see Figure III.31) button and select **Blue to Red Rainbow** (HSV) in the list of **Preset Color Scales**. If it doesn't appear in the list you can Press **Apply** and then **Close**.

Choose Preset ×			
Blue to Red Rainbow		Options to load:	
Presets	Presets	Colors	
Blue to Red Rainbow		✓ Opacities	
		Use preset range	
		Actions on selected:	
		Show current preset in default mode	
		Apply	
		Import	
		Export	
		Remove	
Tip: <click> to select, <double-click> to apply a preset.</double-click></click>		Close	

Figure III.32: Selecting the visualisation colour defaults.

Press the save current display settings values as default button to save the changes. The contour plot of velocity magnitude is now updated for the new colour scale (Figure III.33).



Figure III.33: Contour plot of velocity magnitude.

The contour plot indicates that there is a zone of higher velocity flow defined by the green and red zones surrounded by lower and no-velocity regions in blue. Consistent with the chosen boundary conditions, the maximum velocity is equal to 1.0 m/s at the top, sliding wall and decreases to zero at the other, non-slip walls.

With the HSV colour scale, the contour plot is now clear, with the different levels clearly differentiable,

		$code\_saturne$
EDF R&D	code_saturne version 8.0 tutorial: Shear Driven Cavity Flow	documentation Page $32/37$

but the image looks tessellated. As code\_saturne outputs data at cell centres in the **results** file, in ParaView each mesh cell is painted with a pixel of colour corresponding to the exact value of the variable in the cell. Whilst this cell-data visualisation mode is correct to examine exact values at cell centres, it can yield ragged images unrepresentative of the solution's higher-order spatial accuracy and cannot be used in ParaView to generate vectors and streamline plots. Instead, to produce smoother images, and vector and streamline plots, the cell-data can be interpolated to cell vertices (corners) using the ParaView CellDatatoPointData filter.

Having clicked on **RESULTS\_FLUID\_DOMAIN.case** in the Pipeline Browser to select it, in the top menu bar click on Filters Alphabetical CellDatatoPointData. The new object, which now stores the **RESULTS\_FLUID\_DOMAIN.case** data interpolated to cell vertices, is added in the Pipeline Browser (Figure III.34).

Pip	eline Browser
	builtin:
$\widehat{\gamma_{r'}}$	RESULTS_FLUID_DOMAIN.case
۲	💼 CellDatatoPointData1

Figure III.34: CellDatatoPointData filter added to the Pipeline Browser.

With CellDatatoPointData selected in the Pipeline Browser, click Apply in the Properties tab of the Object Inspector to bring up the smoothed-out contour plot in the ParaView scene viewer.





Picking the different variables to map, either from the **Display** tab in the Object Inspector for the **CellDatatoPointData** object or directly from the drop-down list in the menu bar for contour plots at the top of the SALOME window (see Figure III.35), create contour plots for the X and Y velocity components, as shown in Figure III.36 below. The plots use a **Surface** mode of **Representation**, which can be selected either from the **Display** tab and **Representation** category, or directly from the drop-down list in the contour plot menu bar (see Figure III.35).



Figure III.36: Contour plots of the X (left) and Y (right) velocity components.

As expected, the maximum velocity in the X direction occurs at the top wall, where it nears 1.0 m/s

(Figure III.36 (left)). The locations of the blue (flow in the negative direction) and red (flow in the positive direction) areas in both plots (Figure III.36) indicate that the flow is entrained by the top wall in a clockwise circular motion inside the box. These plots are useful to verify overall velocity magnitudes and patterns but, to get a better visualisation of the flow, one can create 2D glyphs which shows velocity vectors and streamlines which will show fluid particles trajectories.

To create the 2D glyphs, select **CellDatatoPointData** in the Pipeline Browser and click on Filters Common Glyph in the top menu. The Glyph1 object is then added under CellDatatoPointData in the Pipeline Browser. Keep the CellDatatoPointData object selected and visible ( icon in bold next to the object). In the Object Inspector, select the Properties tab. In the Glyph Source category, choose 2D Glyph in the Glyph Type drop-down list. In the Scale category, choose Velocity in the Scale array drop-down list and Scale by Magnitude in the Vector Scale Mode drop-down list (see Figure III.37). Press Apply].

📟 Properties (	Glyph1)			
Glyph Source				
Glyph Type 2D Gly	rph 👻			
Scale				
Scale Array	◆ Velocity			
Vector Scale Mode	Scale by Magnitude 👻			
✓ Rescale Glyphs				
Glyph Data Rang	e			
Minimum	0.00			
Maximum 🥌	0.85			
Glyph Size				
Maximum Glyph Size	0.100000 ×. ©			
Orientation				
Orientation Array	◆ Velocity ▼			
Masking				
Glyph Mode	Uniform Spatial Distribution 👻			
Maximum Number 5000				
Seed	10339			

Figure III.37: Glyph type and Scale Mode.

In the Object Inspector, select the **Display** tab. In the **Coloring** category, choose **Solid Color** and set this color to black by clicking on the **Edit color map** button (see Figure III.38).Press **OK** to finish.



Figure III.38: Glyph coloring.

The result should look like Figure III.39.



Figure III.39: Contour plot of the velocity with 2D glyphs.

To make it possible to create and locate the streamlines with regard to the Cavity, you are going to create a combined image showing the streamlines superimposed on top of the mesh.

First, create the streamlines. With CellDatatoPointData selected in the Pipeline Browser, click on Filters Common Stream Tracer in the top menu. The SteamTracer object is then added under CellDatatoPointData in the Pipeline Browser. Keep the CellDatatoPointData object selected and visible ( icon in bold next to the object). In the Object Inspector, select the Display tab. In the Coloring category, choose Solid Color in the Color by drop-down list. In the Style category, change the Representation to Wireframe and decrease the Opacity to 0.1 (Figure III.40).

🗖 Display (U	nstructuredGridRe	presentatio	on) 🗈 🗈 🤤 🛓
Representation	Wireframe		•
Coloring			
Osolid Color	•		
	🔒 Edit	<u>i</u> 2	
Styling			
Opacity	-		0.100000
Line Width	1.000000		

Figure III.40: Selecting the opacity.

The mesh lines displayed in the **ParaView scene Viewer** should be painted in faint black colour.

Next, select the **StreamTracer1** object in the Pipeline Browser. In the **Object Inspector**, select the **Properties** tab and modify the default settings for **Seeds**. Select **Point Source** for the **Seed Type**, and modify the X and Y coordinates of the seed point to 0.15 and 0.05, respectively. Request 60,000 points (**Number of Points**) and a radius of 1.0 (**Radius**), as shown in Figure III.41 below. Press Apply to validate your changes.

Seeds		
Seed Type Point Source		•
Sphere Parameters		
✓ Show Sphere		
Center 0.15	0.05	0.0049999998882412
Radius 1.0		
Note: Use 'P' to a 'Cent mesh point	er' on mesh or 'C	trl+P' to snap to the closest
Number Of Points 60000		

Figure III.41: Selecting the streamlines settings.

In the **Display** tab, choose **Velocity** and **Magnitude** from the **Color by** drop-down lists of category **Color** and visualise the streamlines coloured by velocity magnitude, on top of the computational mesh (Figure III.42).



Figure III.42: Streamlines coloured by velocity magnitude superimposed on the mesh.

In addition to the general circular motion which had been inferred from the contour plots of velocity components, the streamlines now reveal two recirculation pockets in the bottom corners of the Cavity.

To further compare the calculated data with the available data from [4], create line plots of the X and Y velocities along the Y-axis and X-axis profiles which were set in the code\_saturne GUI. Using the menu File Open ParaView File, import both the XVel\_YaxisCentreLine.csv and the YVel\_XaxisCentreLine.csv files. To read in the data and ensure that the velocities are correctly interpreted as real numbers by ParaView, in the Properties tab, define the Field Delimiter Characters as ',': a comma. Also, select Merge Consecutive Delimiters (Figure III.43).

EDF R&D	code_saturne version 8.0 tutori Shear Driven Cavity Flow	al:	code_saturne documentation Page 36/37
	- XVel_YaxisCenterLine.csv YVel_XaxisCenterLine.csv	v	
	Properties	6	)
	Apply ØReset Delete	?	
	Search (use Esc to clear text)	200	
	Properties (YVel_XaxisCenterLine.csv)		
	✓ Detect Numeric Columns		
	✓ Use String Delimiter		
	✓ Have Headers		
	Field Delimiter		

Figure III.43: Specifying the velocity profiles.

Add Tab Field Delimiter

 Merge Consecutive Delimiters

Go to the **Information** tab (if it doesn't appear, you can enable it by selecting <u>View</u><u>Windows</u>) <u>Information</u>) and verify that all the data arrays are read in as **double**. The steps to create and label the graphs are similar to those already detailed for the graphs of the **monitoring points** data and they are not repeated here.

For direct comparisons with the plots from [4], for the graphs of X velocity, plot the Y coordinate as a function of Velocity[0]. For the graphs of Y velocity, plot Velocity[1] as a function of the X coordinate. The reference data provided in [4] for the fine mesh results (129x129) is listed in Section 4. The data can be copied to a .csv file which is then imported in ParaView at the same time as the XVel\_YaxisCentreLine.csv and the Y\_VelXaxisCentreLine.csv files. Set the plots so that the code\_saturne and the reference results are visualised in the same view, and display the reference points with a marker rather than a line. Each data set can then be compared directly with the reference results (Figure III.44).



Figure III.44: Comparison between code\_saturne (solid lines) and reference [4] results (circles). Y coordinate versus X velocity (left) and Y velocity versus X coordinate (right).

Overall, good agreement is obtained with the reference results [4], even though the results with code\_saturne were obtained on a coarser mesh. Running on a finer mesh would make it possible to capture the velocity extremas with increased accuracy.

#### 3 References

- [1] www.salome-platform.org
- 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] U. GHIA, K.N. GHIA, and C.T. SHIN, *High-Re solutions for incompressible flow using the Navier-Stokes equations and a multigrid method*, Journal of Comp. Phys., Vol. 48, pp. 387-411, 1982.
- U. AYACHIT, *The ParaView Guide: A Parallel Visualization Application*, Kitware, 2015, ISBN 978-1930934306

## 4 Reference Data from [4]

To make it easier to replicate the comparative plots of the code\_saturne and reference results, the data from [4] (Tables I and II) at Re = 1000 is reproduced in Figure III.4 below. The reference results listed in [4] were obtained on a fine, 129x129 mesh.

Y (m)	$\mathbf{U_x}~(\mathbf{m/s})$	X (m)	$\mathbf{U_y}~(\mathbf{m/s})$
1.00000	1.00000	1.00000	0.00000
0.97660	0.65928	0.96880	-0.21388
0.96880	0.57492	0.96090	-0.27669
0.96090	0.51117	0.95310	-0.33714
0.95310	0.46604	0.94530	-0.39188
0.85160	0.33304	0.90630	-0.51550
0.73440	0.18719	0.85940	-0.42665
0.61720	0.05702	0.80470	-0.31966
0.50000	-0.06080	0.50000	0.02526
0.45310	-0.10648	0.23440	0.32235
0.28130	-0.27805	0.22660	0.33075
0.17190	-0.38289	0.15630	0.37095
0.10160	-0.29730	0.09380	0.32627
0.07030	-0.22220	0.07810	0.30353
0.06250	-0.20196	0.07030	0.29012
0.05470	-0.18109	0.06250	0.27485
0.00000	0.00000	0.00000	0.00000

Table III.4: X velocity versus Y at the vertical mid-plane and Y velocity versus X at the horizontal mid-plane. Data reproduced from [4]. Re = 1000. 129x129 mesh.