



POLITECNICO DI MILANO

MSC MECHANICAL ENGINEERING
INTERNAL COMBUSTION ENGINE GROUP

MASTER THESIS

Direct Efficiency Evaluation of an Automotive Radial Turbine

Advisor:

Prof. Eng. Gianluca Montenegro

Co-Advisor:

name surname

Master Thesis Work of:

Edoardo Mancuso ID: 852853

Academic Year 2016/2017

Abstract

The following work places itself in the field of **numerical modeling applied to internal combustion engines** research activities. The study has been carried out along the **ICE Group** based at Politecnico of Milan University.

The aim is to develop new methodologies to evaluate turbomachines performances. In particular, focus has been placed on the flow dis-homogeneity downstream of a typical automotive turbine, that leads to a wrong measurement of the iso-entropic efficiency. The reference experimental work comes from **ICEG** (Internal Combustion Engines Group) based at **Scuola Politecnica dell'Università di Genova**. After having performed all the needed measures in the standard usual configuration the results showed that, with a more sophisticated sampling, efficiency could grow up to 5% with respect to reference values.

The subsequent campaign was devoted to the evaluation of different **flow-conditioner**, with the aim of reducing the dis-uniformities and at the same time not introducing too much pressure losses. This permitted to correctly map the machine just with a few local sampling, without the need of huge measurement campaigns. Our aim is to develop a **numerical model** able to predict accurately the flow-physic inside such a configuration. Once the correspondence between experimental and numerical models will be achieved, it will be possible to efficiently analyze the flow dynamic inside this components with the aim to provide an optimized design.

Moreover thanks to CFD we will be able to base the efficiency computation on an actual adiabatic transformation, hence avoiding the effect of thermal losses that, usually, represent one of the biggest source of uncertainty.

Contents

1	Introduction to the Test Case	11
1.1	Problems related to efficiency evaluation	11
1.1.1	Definition	11
1.1.2	Flow dis-uniformities related issues	11
1.1.3	Heat exchange related issues	11
1.2	Design of Experiment	14
1.2.1	Experimental Apparatus	14
1.2.2	Measurement Technique	14
1.2.3	Test conditions	17
2	Computational Fluid Dynamics	18
2.1	Navier-Stokes Equations	18
2.2	Solving Equations	19
2.3	Spatial Discretization	19
2.3.1	Finite Volume Method	19
2.3.2	Mesh Quality Indexes	20
2.4	Reynolds Average Navier-Stokes Equations	21
3	Preprocessing	23
3.1	Domain Definition	23
3.1.1	Scroll	23
3.1.2	Rotor	23
3.1.3	Discharge	23
3.1.4	Honeycomb Decoupler	25
3.2	Meshing strategy	26
3.2.1	cfMesh	26
3.2.2	cartesianMesh	26
3.2.3	Meshing Result	28
3.2.4	Rotating zone definition	31
3.2.5	Porous zone definition	31
3.2.6	Mesh merging	32
3.3	Boundary condition	33
3.3.1	Temperature	33
3.3.2	Pressure	33
3.3.3	Velocity	33
3.3.4	Turbulence	34
3.4	Rotor Rotational Speed	34
3.4.1	Multiple Reference Frame Approach	34
3.5	Porous Zone Set-up	34

3.5.1	Darcy-Forchheimer model	35
3.5.2	Coefficient set-up	35
3.5.3	Turbulence treatment	35
3.6	Evaluating decoupler performances	37
4	Solution Technique	38
4.1	The SIMPLE algorithm	38
4.1.1	The Pressure Poisson Equation	38
4.1.2	Algorithm Steps	39
4.1.3	MRF equations	40
4.1.4	MRFRhoSimpleFoam	41
4.1.5	rhoPorousSimpleFoam	42
4.2	Numerical Schemes	43
4.2.1	Finite Volumes schemes	43
4.2.2	Finite Volumes solutions	44
5	Standard duct	46
5.1	Turbine Map	46
5.2	Adiabatic Case	48
5.2.1	Mesh Statistics	48
5.2.2	yPlus	49
5.2.3	Residuals	49
5.2.4	Graphic Post Processing	50
5.2.5	Numerical Results	51
5.2.6	Meridional and tangential velocity maps	52
5.3	Fixed Wall Temperature	54
5.3.1	Residuals	54
5.3.2	Graphic Post Processing	54
5.3.3	Numerical Results	55
5.4	Wall Heat Flux	57
5.4.1	Residuals	57
5.4.2	Graphic Post Processing	58
5.4.3	Numerical Results	59
5.5	Uneven Wall Heat Flux	60
5.5.1	Residuals	60
5.5.2	Graphic Post Processing	61
5.5.3	Numerical Results	62
5.6	Grid sensitivity analysis	64

6	Honeycomb Expansion Chamber	66
6.1	Porous Coefficient Set Up	66
6.2	Turbulence Analysis	67
6.2.1	Meshing	67
6.2.2	Results	67
6.3	Results: first operating point, adiabatic walls	70
6.3.1	Mesh Statistics	70
6.3.2	Residuals	71
6.3.3	Numerical Results	71
6.4	Results: first operating point, addition of heat losses	74
6.4.1	Residuals	74
6.4.2	Numerical Results	74
6.5	Results: second operating point	76
6.5.1	Residuals	76
6.5.2	Numerical Results	76
6.6	Grid sensitivity analysis	77
6.7	Decoupling performances	79
6.7.1	First operating point	80
6.7.2	Second operating point	82
6.8	Influence of turbulence treatment	83
6.9	Influence of expansion chamber diameter	84
6.9.1	Results	84
6.10	Single honeycomb matrix	86
6.10.1	Results	86
7	Diffuser decoupler	89
7.1	Diffusing related issues	89
7.2	Geometry and meshing	90
7.3	Results	91
8	Asymmetrical diffuser decoupler	93
8.1	Geometry and meshing	93
8.2	Results	94
9	Honeycomb mixer decoupler	97
9.1	Geometry and meshing	97
9.2	Results	99
9.3	Mixer modification	101

10	Pyramidal honeycomb decoupler	103
10.1	Geometry and meshing	103
10.2	Results	104
11	Hexagonal honeycomb decoupler	108
11.1	Geometry and meshing	108
11.2	Results	109
11.3	Results: second operating point	112
11.4	Results: smaller hexagonal matrix	113
12	Honeycomb alternatives	115
12.1	Macro Honeycomb	115
12.1.1	Geometry and meshing	115
12.1.2	Results	116
13	Final remarks	119
14	Bibliography	120
A	Case Setup Dictionaries	121
B	Boundary Conditions	125
B.1	MRFrhoSimpleFoam	125
B.2	Porous flow-metering	128
B.3	Porous turbulence evaluation	129
B.4	fvOptions porous media	130
C	Numerical Schemes	131
C.1	MRFrhoSimpleFoam	131
C.2	Porous turbulence evaluation	133

List of Figures

1	Simplified representation of characteristics turbochargers heat fluxes	12
2	Enthalpy-entropy diagram of the expansions process	13
3	Measurement sections scheme	14
4	Scheme of the experimental configuration	15
5	Front and side view of the 3 holes probe	15
6	Section discretization	16
7	3D finite volume representation	19
8	2D example of non-orthogonality between two elements	20
9	Representation of skewness between two neighbors cells	21
10	Volute and Rotor CAD files	24
11	Discharge CAD file	24
12	Honeycomb decoupler CAD file and section	25
13	surface and edges files used to define further refinement	27
14	Volute Mesh result	28
15	Rotor Mesh result	29
16	Discharge Mesh result	30
17	Porous cell zones	31
18	Merged Domain	32
19	AMI coupled patches	32
20	2D sketch of the domain used for turbulence analysis	36
21	3D representation of the domain used for turbulence analysis	36
22	Turbine Map	47
23	Total temperature contours,cross section and measurement plane, standard duct adiabatic	50
24	Axial velocity contours and vectors, cross section and measurement plane, standard duct adiabatic	50
25	Total temperature, standard duct, adiabatic walls	51
26	Axial velocity, standard duct, adiabatic walls	52
27	Axial velocity contours at measurement section, numerical (left) experimental(right)	53
28	Tangential velocity, standard duct, adiabatic	53
29	Total temperature contours, cross section and measurement plane, standard duct fixed wall temperature	54
30	Axial velocity contours and vectors, cross section and measurement plane, standard duct fixed wall temperature	55
31	Total temperature, standard duct, fixed wall temperature 330 K	55
32	Axial velocity, standard duct, fixed wall temperature 330 K	56
33	Total temperature contours, cross section and measurement plane, standard duct wall heat flux	58

34	Axial velocity contours and vectors, cross section and measurement plane, standard duct wall heat flux	58
35	Total temperature, standard duct, wall heat flux 2950 W/m ²	59
36	Axial velocity, standard duct, wall heat flux 2950 W/m ²	60
37	Total temperature contours, cross section and measurement plane, standard duct, uneven wall heat flux	61
38	Axial velocity contours and vectors, cross section and measurement plane, standard duct, uneven wall heat flux	61
39	Total temperature, standard duct, uneven wall heat flux	62
40	Axial velocity, standard duct, uneven wall heat flux	63
41	Total temperature and axial velocity, grid independence analysis standard duct	65
42	Experimental and numerical honeycomb expansion chamber flow meter	66
43	Honeycomb channel mesh and cross section detail	67
44	Turbulent kinetic energy and specific frequency, whole domain	68
45	Turbulent kinetic energy and specific frequency, half channel	68
46	κ and ω contours, first measurement section	68
47	κ and ω contours, second measurement section	69
48	Total temperature, honeycomb expansion chamber, adiabatic	71
49	Axial velocity, honeycomb expansion chamber, adiabatic	72
50	Tangential velocity, honeycomb expansion chamber, adiabatic	73
51	Axial velocity contours and vectors, cross section and measurement plane, honeycomb expansion chamber, adiabatic	73
52	Total temperature and axial velocity, honeycomb expansion chamber, heat losses addition	75
53	Total temperature, honeycomb expansion chamber, second operating point	76
54	Axial velocity, honeycomb expansion chamber, second operating point	77
55	Total temperature and axial velocity, honeycomb expansion chamber, grid independence	78
56	Sampled Points	79
57	Graphical post-processing, honeycomb expansion chamber, axial velocity contours	80
58	Efficiency, expansion chamber standard (top) and refined (bottom) grids	81
59	Efficiency, expansion chamber, n_{red} 5500 rpm \sqrt{K}	82
60	Turbulent kinetic energy and specific frequency contours, without (top) and with source (bottom)	83
61	Reduced (top) and increased (bottom) expansion chamber domains	84
62	Efficiency, reduced(top) and increased (bottom) expansion chamber	85
63	Single honeycomb expansion chamber domain	86
64	Graphical post processing, axial velocity contours, single HC expansion chamber	87

65	Efficiency, single HC, standard (top) and refined (bottom) grid	88
66	Radial Equilibrium	89
67	3D velocity vector inside decoupler's casing	89
68	Technical drawing of the diffuser geometry	90
69	Mesh of the diffuser discharge	90
70	Graphical post-processing, diffuser decoupler, axial velocity contours	91
71	Graphical post-processing, honeycomb diffuser, axial velocity contours	92
72	Efficiency, honeycomb diffuser	92
73	Technical drawing, asymmetrical diffuser	93
74	Mesh, asymmetrical diffuser	93
75	Graphical post-processing, asymmetrical diffuser, axial velocity contours	94
76	Cross section detail of the refined grid	95
77	Efficiency, asymmetric diffuser, standard grid (top) and refined (bottom)	96
78	Graphical post-processing, standard duct, axial velocity contours	97
79	Detail of the mixing fins and their meshing	98
80	3D drawing, honeycomb mixer decoupler	98
81	Graphical post-processing, mixer honeycomb decoupler, axial velocity contours	99
82	Efficiency, honeycomb mixer	100
83	Grid generation on the new domain	101
84	Graphical post-processing, modified mixer honeycomb decoupler, axial velocity contours	101
85	Efficiency, honeycomb mixer modified diffuser	102
86	Front Back and section view of the generated geometry	103
87	Grid detail, pyramidal honeycomb	104
88	3D vectors representation, velocity magnitude, pyramidal HC	104
89	Axial cross section, velocity magnitude contours and vectors, pyramidal HC	105
90	Efficiency, pyramidal HC decoupler	105
91	Grid detail, refined pyramidal honeycomb	106
92	Axial cross section, axial velocity contours and vectors, refined pyramidal HC	106
93	Efficiency, pyramidal HC decoupler, refined grid	107
94	Front and Back view of the generated geometry	108
95	Grid Detail	108
96	3D vectors representation, velocity magnitude, hexagonal HC	109
97	Multiple Sections, velocity magnitude contours and vectors, hexagonal HC	109
98	Efficiency, hexagonal HC decoupler, standard grid	110
99	Efficiency, hexagonal HC decoupler, refined grid	111
100	Efficiency, hexagonal HC decoupler, $n_{red} 5500 rpm \sqrt{K}$	112
101	3D vectors representation, velocity magnitude, small hex HC	113
102	Multiple Sections, velocity magnitude contours and vectors, small hex HC	113
103	Efficiency, small hex HC decoupler	114
104	3D geometry representation, standard duct and asymmetrical diffuser	115

105	Grid, standard duct and asymmetrical diffuser	116
106	Grid, standard duct and asymmetrical diffuser	116
107	Efficiency, standard duct macro honeycomb	117
108	Efficiency, macro honeycomb diffuser	118

List of Tables

1	experimental conditions	17
2	Volute mesh sizing (element's edge length)	28
3	Rotor mesh sizing (element's edge length)	29
4	Discharge mesh sizing (element's edge length)	30
5	Turbine Map BC	46
6	Grid features, standard duct discharge	48
7	Average y^+ values, standard duct discharge	49
8	Initial residual values, standard duct, adiabatic	49
9	Initial residual values, standard duct, fixed wall temperature	54
10	Initial residual values, standard duct, wall heat flux	57
11	initial residual values, standard duct, uneven wall heat flux	60
12	Refined volute Mesh Sizing (element's edge length)	64
13	Refined rotor mesh sizing (element's edge length)	64
14	Refined discharge mesh sizing (element's edge length)	64
15	Initial residual values, porous coefficient set-up	66
16	Grid Features, honeycomb turbulence analysis	67
17	Initial residuals, honeycomb turbulence analysis	67
18	κ and ω average values, turbulence analysis	69
19	Grid feature, honeycomb expansion chamber	70
20	Average y^+ values, honeycomb expansion chamber domain	70
21	Initial residuals, honeycomb expansion chamber adiabatic	71
22	Initial residuals values, honeycomb expansion chamber, first operating point	74
23	Initial residuals values, honeycomb expansion chamber, second operating point	76
24	Performances indexes, honeycomb expansion chamber, $n_{red} 4750 rpm \sqrt{K}$. .	80
25	Performances indexes, honeycomb expansion chamber, $n_{red} 5500 rpm \sqrt{K}$.	82
26	Performances indexes, honeycomb expansion chamber with turbulence sources	83
27	Initial residuals values, reduced and increased expansion chamber volume .	84
28	Performances indexes, reduced and increased expansion chamber	86
29	Initial residuals values, single honeycomb expansion chamber	87
30	Performances indexes, single HC expansion chamber	87
31	Initial residuals values, honeycomb diffuser	91
32	Performances indexes, honeycomb diffuser	91
33	Initial residuals values, honeycomb diffuser	94

34	Performances indexes, asymmetrical honeycomb diffuser	95
35	Performances indexes, honeycomb mixer	99
36	Initial residuals values, honeycomb mixer	99
37	Initial residuals values, honeycomb modified mixer	102
38	Performances indexes, honeycomb mixer	102
39	Initial residuals, pyramidal honeycomb	104
40	Performances indexes, pyramidal honeycomb	105
41	Initial residuals values, pyramidal HC refined	106
42	Performances indexes, pyramidal honeycomb, refined grid	107
43	Initial residuals values, hexagonal HC	110
44	Performances indexes, hexagonal HC decoupler	111
45	Performances indexes, hexagonal HC decoupler, $n_{red} 5500 rpm\sqrt{K}$	112
46	Initial residuals values, smaller hex HC	114
47	Performances indexes, hexagonal honeycomb	114
48	Initial residuals values, macro HC	117
49	Performances indexes, macro honeycomb	118

1 Introduction to the Test Case

1.1 Problems related to efficiency evaluation

1.1.1 Definition

Usually when talking about turbines, the performance are evaluated in terms of total to static iso-entropic efficiency (η_{isTS}) [1]. We need to take into account only the static energetic contribution at the outlet section since the remaining dynamic content is then lost to the ambient. The generic expression is reported in 1:

$$\eta_{isTS} = \frac{h_{Tin} - h_{Tout}}{h_{Tin} - h_{out_{is}}} \quad (1)$$

In case of an adiabatic transformation for an ideal gas with constant specific heat ratio γ , we could write:

$$\eta_{isTS} = \frac{T_{Tin} - T_{Tout}}{T_{Tin} - T_{out_{is}}} = \frac{T_{Tin} - T_{Tout}}{T_{Tin} \left[1 - \left(\frac{p_A}{p_{Tin}} \right)^{\frac{\gamma-1}{\gamma}} \right]} = \frac{T_{Tin} - T_{Tout}}{T_{Tin} \left[1 - \left(\frac{1}{\beta_{TS}} \right)^{\frac{\gamma-1}{\gamma}} \right]} \quad (2)$$

where with β_{TS} we express the total to static expansion ratio.

1.1.2 Flow dis-uniformities related issues

From equation 2, it is already possible to point out that, especially in an internal combustion environment where we face modest expansion ratios, the total temperature jump plays the fundamental role in the determination of this efficiency. As a matter of fact, even a small measurement error, reflects heavily on the performance estimation.

Usually, to reduce the kinetic head loss, turbines blades are designed to discharge the flow axially [2]. Since this latter angle is function of the ratio between the mass-flow and the rotational speed, the previous statement could be taken as valid only in a few design conditions, whereas in our case, we have to cover a very wide map of operating points. This leads to the fact that, most of the time, discharged flow field will feature an important swirling component.

The general approach adopted to build turbines map [3], consists in measuring the outlet total temperature through three different probes, usually staggered at 120° one from each other and then average the results. The problem is, as we will see later on, that such a configuration isn't enough to obtain accurate results.

1.1.3 Heat exchange related issues

As pointed out previously, the calculation used to compute the efficiency is based on an adiabatic transformation. As we can see in the simplified representation in fig. 1, in a real application, we're facing lots of heat transfer related phenomena.

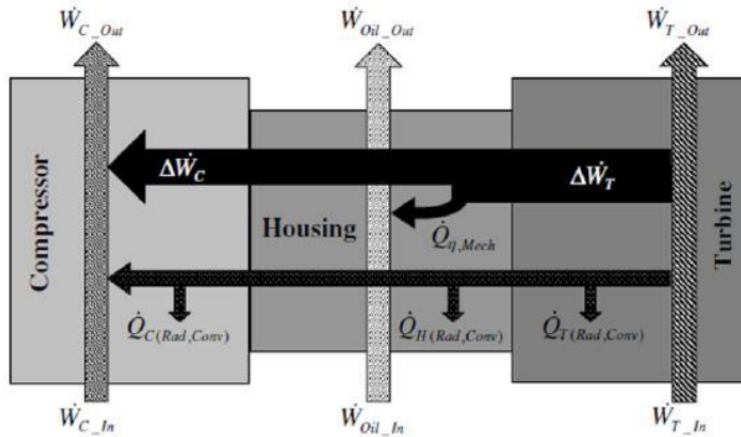


Figure 1: Simplified representation of characteristics turbochargers heat fluxes

The total exchanged power is given by the sum of three contributions [4]:

- Conduction through the casing;
- Irradiation from the turbine casing towards the compressor's;
- Conduction between the gases and the lubrication fluid.

Focusing on the turbine side, the flow exits at a temperature lower with respect to the adiabatic correspondent: heat is transferred to the case and to the lubrication system. The biggest issue is related to the impossibility of distinguishing heat and work exchange phenomena:

- Temperature drop due to expansion and heat loss;
- Heat transferred to lubrication fluid and heat generated by viscous losses in the bearings.

In figure 2 it is reported a enthalpy-entropy diagram of the expansion process. It is clear to see that the total enthalpy (temperature) drop in the real condition (point 2) is grater than in the adiabatic case (point 2_{adb}), since it includes also the heat rejected both in expansion and in the outlet duct).

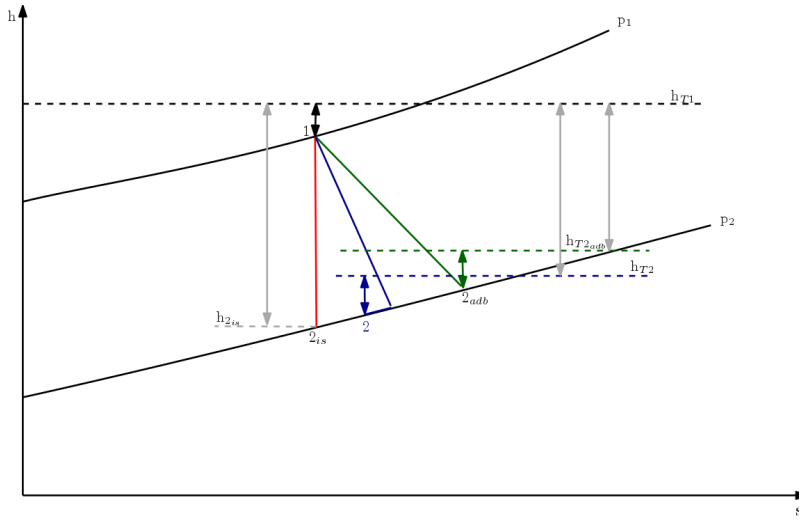


Figure 2: Enthalpy-entropy diagram of the expansions process

As a matter of facts, Serrano [5] reports values of efficiency higher than unity for operating points corresponding to small rotational speeds and high total inlet temperature. Since low speed means also low flow-rate, the effect of heat exchange becomes more and more relevant, though leading to non physical performance figures. This is the main reason why the experiments, onto which this work is based on, are mostly carried out in cold-flow condition where the relative importance of the heat exchange on the total enthalpy jump drops consistently.

1.2 Design of Experiment

Our numerical model refers to the compressed air test bench located at **DIME**(Dipartimento di Ingegneria Meccanica, Energetica, Gestionale e dei Trasporti) in Genova.

1.2.1 Experimental Apparatus

The apparatus permits to develop experiments on automotive turbochargers both in steady state and transient conditions. Air is supplied by three electro actuated screw compressors that could generate up to 0.65 kg/s mass-flow-rate. The test bench could work both in cold and hot flow condition thanks to an electrical heater station. In the cold setup, air, is nevertheless heated in order to avoid possible freezing on the turbine blades due to strong expansion ratios.

The following sub systems could be identified:

- Compression station;
- Air circuit;
- Lubrication system;
- Control Room;

1.2.2 Measurement Technique

Referring to 3 where we could see two different section, 4N (near) and 4F (far), placed respectively at 5 and 11.5 diameters ($\phi = 54\text{mm}$) downstream of the turbine discharge section.

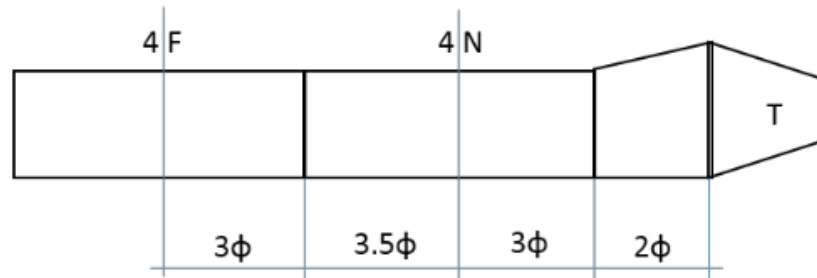


Figure 3: Measurement sections scheme

For this kinds of experiments, the standard measurement setup, is composed of three thermocouples for the evaluation of total temperature and a pressure ring for the static pressure. In addition to the standard set-up, an ad-hoc 4 setup as been built.

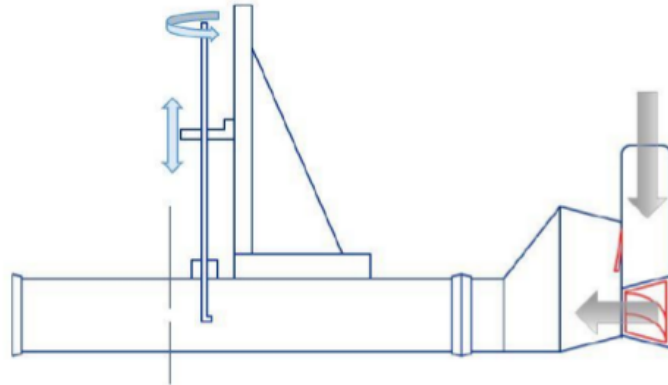


Figure 4: Scheme of the experimental configuration

This particular configuration permits, through 2 servo motors, to displace the measurement device along the section. Data acquisition comes through a three holes pneumatic probe 5, equipped also of a thermocouple.

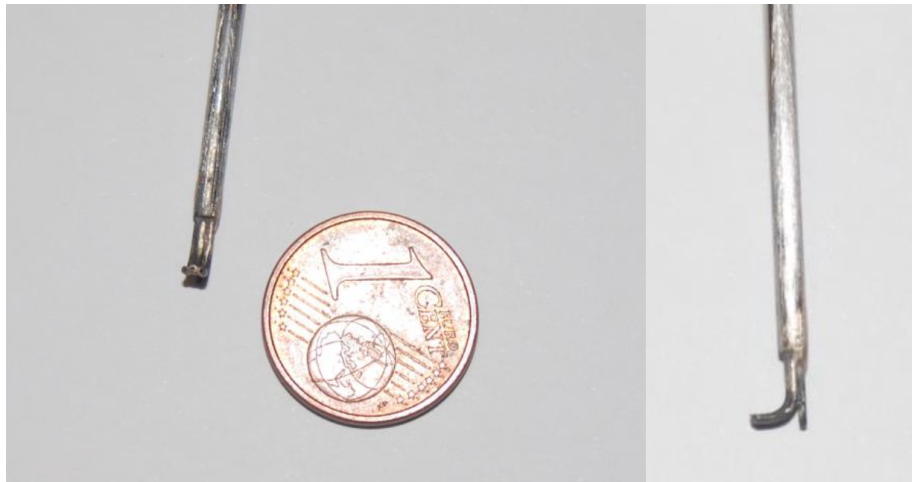


Figure 5: Front and side view of the 3 holes probe

The central hole (A) measures the total pressure through a piezoresistive transducer whereas, the two lateral holes (B,C), capture the difference in the static thanks to a differ-

ential pressure transmitter. The use of a three way electro-valve permits to compute both the delta A-B or C-B, so to obtain the dynamic contribute of the total pressure, hence the velocity. The radial measurements position are chosen 5 mm one from each other 6.. Then the device is rotated of 120° in both clock and anti clock wise direction so to obtain three different values at each probe depth. This means 25 acquisition points of static and total pressure moreover, the usage of this peculiar configuration permits to evaluate also the velocity field. Through these measurements it is then possible to have a value of the mass-flow-rate in each discretized area 3. This lead to the possibility of obtaining an efficiency value that do not come from a simple arithmetic average but from a weighted one 5.

$$\dot{m} = \sum \rho_i c_{i_{ax}} A_i \quad (3)$$

Where c_{ax} is the mean value of all the axial velocity in each cross section.

Now to directly evaluate the efficiency of the machine, the enthalpy fluxes must be computed. For what concern the inlet section, assuming a uniform flow:

$$H_{T3} = \dot{m}_3 c_{p3} T_{T3} \quad (4)$$

Shifting toward the outlet section, we could use the same method as the one depicted above for the massflow:

$$H_{T4} = \sum \dot{m}_i c_{p4} T_{T4} \quad (5)$$

So the total-to-static iso-entropic efficiency (under the adiabatic transformation hypothesis):

$$\eta_{TS} = \frac{H_{T3} - H_{T4}}{H_{T3} - H_{4is}} = \frac{H_{T3} - H_{T4}}{H_{T3} [1 - (\frac{1}{\beta_{TS}})^{\frac{\gamma-1}{\gamma}}]} \quad (6)$$

where β_{TS} is the total to static expansion ratio and γ the ratio between the specific heats.

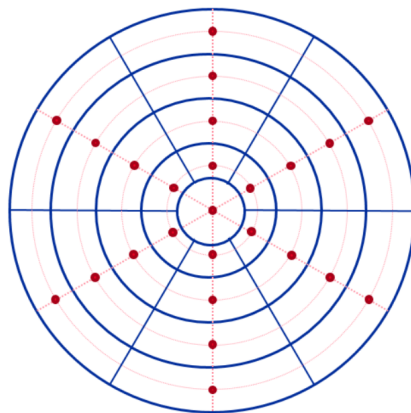


Figure 6: Section discretization

1.2.3 Test conditions

As briefly stated before, both hot and cold conditions were evaluated, where with hot we're referring to a total inlet temperature T_{in} of $80^{\circ}C$, whereas hot stays for T_{in} equal to $300^{\circ}C$.

As the similitude theory state, imposing a certain pressure ratio is equal to impose a certain reduced mass-flow rate, if the reduced rotational speed remains constant. Thus adjusting the actual angular velocity of the rotor to the total inlet temperature, permits to theoretically obtain the same pressure ratio.

Firstly, measures have been taken in the configuration without flow conditioner. These "numbers" will provide the references to see if that, through a decoupler, we will be able to create an even flow in order to efficiently measure the efficiency without the need of sophisticated radial acquisitions.

The experimental trials have been carried out in the following range of initial conditions:

Variable	MU	Values
T_{in}	$^{\circ}C$	80 - 300
n_{rid}	$rpm K^{0.5}$	4000-4750-5500
β_{TS}		1.27 1.38 1.4 1.5

Table 1: experimental conditions

Our work, will basically be structured in the same way. First we will build the model of the machine without conditioners so to prove the numerical model robust and able to actually reproduce the experimental values. The most promising kind of decoupler, namely the honeycomb, will be added later and investigated in order to prove its validity and if possible to improve it.

2 Computational Fluid Dynamics

Computational Fluid Dynamic is based on the formulation of conservation laws.

2.1 Navier-Stokes Equations

Three equations can completely determine the behavior of the system [6].

- Conservation of mass: **continuity equation** 7;
- Conservation of momentum: **momentum equation** 8;
- Conservation of energy: **energy equation**; 9;

In particular, when this set of equations are applied to viscous flow, they're also known as *Navier-Stokes*.

$$\frac{\partial \rho}{\partial t} + \vec{\nabla} \cdot (\rho \vec{U}) = 0 \quad (7)$$

$$\frac{\partial (\rho \vec{U})}{\partial t} + \vec{\nabla} \cdot (\rho \vec{U} \vec{U} + p \vec{I} - \vec{\tau}) d\Omega = \rho \vec{f}_e \quad (8)$$

$$\frac{\partial \rho E}{\partial t} + \vec{\nabla} \cdot (\rho H \vec{U} - k \vec{\nabla} T - \vec{\tau} \cdot \vec{U}) = \rho \vec{f}_e \cdot \vec{U} + q_H \quad (9)$$

At this point it is still needed the **constitutive law**, in our case air is treated as a perfect gas 10. As a matter of facts temperature is computed from the energy equation and then the link with the momentum and mass equations pass through the ideal gas law, where the density ρ is coupled to the pressure and temperature.

$$\frac{p}{\rho} = RT \quad (10)$$

The last step is the definition of properties such as **viscosity** μ and **specific heat capacity** C_p . This latter quantities are all specified through the `thermophysicalProperties` dictionary 3. In particular, viscosity is computed through Sutherland's Formula 11:

$$\mu = \frac{1.45 T^{1.5}}{T + 110} 10^{-6} \quad (11)$$

Whereas the specific heat at constant pressure comes from a fifth order polynomial where the coefficients are taken from `janaf` table considering a mixture of 21% O_2 and 79% N_2

2.2 Solving Equations

The solution variables of the **NS** equations are:

- density ρ
- momentum $\rho\vec{U}$
- energy $\rho e(\rho h)$

The continuity equation is coupled to the momentum, the rate of change of the density depends on the momentum divergence $\rho\vec{U}$.

The non-linear convection term in the momentum equation $\nabla \cdot (\rho\vec{U}\vec{U})$ provides the wealth of interaction in fluid flows whereas the diffusion term contains viscous effects.

Looking at the energy equation, the diffusion term is given in terms of temperature T . The coupling to the rest of the system is weaker since it comes through the equation of state: T influences ρ and \vec{U} . It has to be noted that no convection-diffusion equation are available for the pressure but, its gradient appears in the momentum equation, thus the pressure field needs to be calculated in order to be able to solve these equations.

Since we're dealing with a compressible flow $\frac{\partial \rho}{\partial t} \neq 0$ the continuity equation can be used to compute density. Temperature follows from the enthalpy equation hence, pressure comes from the equation of state: $p = p(\rho, T)$.

2.3 Spatial Discretization

2.3.1 Finite Volume Method

Computational fluid dynamics is mostly based on the so called finite volume approach. The equations depicted in the previous section are impossible to be solved on the whole domain as a single volume, thus the need of discretization into a finite number of control volumes which constitute the mesh.

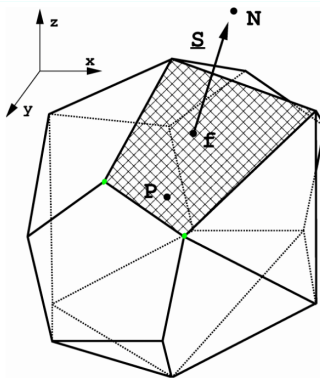


Figure 7: 3D finite volume representation

In fig.7 we could see a general representation where \mathbf{P} indicates the cell centroid that is the point into which the values of the field are stored. \mathbf{f} highlight the face center. With \mathbf{N} we represent the centroid of a general neighbor element. Usually the fields values are assumed to vary linearly within each cell, so a linear interpolation is performed to compute values at face centers.

The partial differential equations governing the problem could also be written in an integral form, then they could be applied to each single control volume and solved. Referring to Φ as the general transported variable, we could write [7]:

$$\int_S \Phi \vec{U} \cdot \vec{n} dS = \int_S \Gamma \vec{\nabla} \Phi \cdot \vec{n} dS + \int_V q_\Phi dV \quad (12)$$

Where the first term represents the convection of the quantity, whereas the latter is composed by diffusion and (an eventual) source. Once the equations coming from all the control volumes are summed up, the internal surface integrals cancel out. Thus global continuity is built into the method. The problem is though that, since integral represent an infinitesimal summation, the can't be computed numerically thus approximations must be introduced. We will deepen this aspect later on when talking about the resolution method of our problem.

2.3.2 Mesh Quality Indexes

Since these elements actually represent the domain of our problem, we have to make sure to produce a great quality grid, able to minimize the error introduced with the discretization. Usually solution is strongly dependent on the spacing, for example the medium element length. It is clear to see that with smaller elements a better approximation could be achieved, but more cells means also higher computational effort so we will need to find the best trade-off. Another important quality aspect is identified in non-orthogonality.

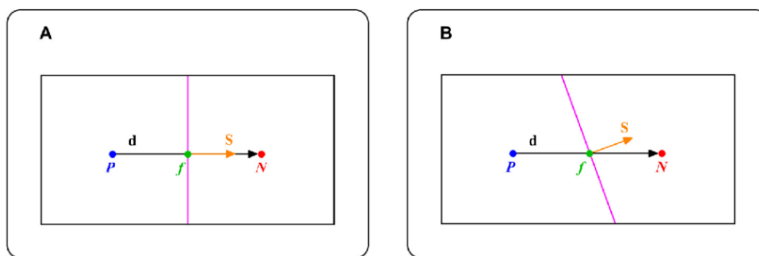


Figure 8: 2D example of non-orthogonality between two elements

In fig.8 we could see a graphical representation of this important index. If we call \vec{S} the normal to cell P east face (pink line), non-orthogonality is identified by the angle between

this vector and ideal line connecting the two neighbors centroid. In case A it is equal to 0 whereas in B it is not. Since gradients (diffusion terms) are evaluated along the cell centers connecting line, if we're in presence of non-orthogonality accuracy is penalized. As a matter of facts, it is actually impossible to keep the value equal to zero especially in case of complex geometries. Thus maximum local values up to 55-65° are accepted with corrections introduced in the calculations.

Last feature we're going to report is skewness ψ .

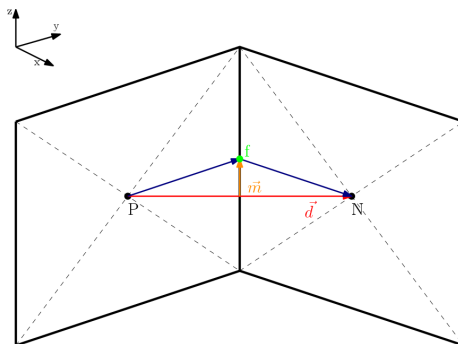


Figure 9: Representation of skewness between two neighbors cells

$$\psi = \frac{|\vec{m}|}{|\vec{d}|}$$

In case of $\psi \neq 0$, hence $|\vec{m}| \neq 0$, the cell centers connecting line does not pass through the actual face center f, the distance used for the interpolation procedure is wrong thus even in this case accuracy is penalized.

2.4 Reynolds Average Navier-Stokes Equations

As in the majority of the real life cases, the hypothesis of a laminar flow could not be made, we have to deal with **turbulence**.

Turbulent flow is a fluid regime characterized by chaotic, stochastic property changes, high momentum convection and rapid variation of pressure and velocity in space and time. Starting from the idea that any property ϕ can be split into a mean $\bar{\phi}$ and a fluctuating ϕ' contributions, the three balances seen before could be re-written and then averaged over time. The results could be summarized considering the effect of the turbulence through additional transport terms that enhance diffusivity and mixing rate [8]:

- Reynolds Stresses;
- Turbulent Heat Flux;

Unfortunately, Reynolds decomposition, does not give any information on the way the fluctuating terms can be expressed, thus the need of a closure problem arise.

The reference model in case of turbo-machinery fluid-dynamics is the $\kappa\omega$ -SST. It is based on 3 fully modeled balances [9]:

- κ : turbulent kinetic energy balance (13);
- ϵ : specific dissipation rate balance (14);
- ω : specific turbulent frequency (15).

$$\frac{\partial \rho \kappa}{\partial t} + \frac{\partial \rho \kappa u_i}{\partial x_i} = \frac{\partial}{\partial x_i} \left(\left(\mu + \frac{\mu_t}{\sigma_t} \right) \frac{\partial \kappa}{\partial x_j} \right) + P_k + P_b - \rho \epsilon - Y_m + S_k \quad (13)$$

$$\frac{\partial \epsilon}{\partial t} + \vec{V} \cdot \nabla \epsilon = C_{\epsilon 1} \frac{\epsilon}{\kappa} : \nabla \vec{V} - C_{\epsilon 2} \frac{\epsilon^2}{\kappa} + \nabla \cdot \left(\left(\frac{\mu}{\rho} + \frac{\mu_T}{\rho \sigma_\epsilon} \right) \nabla \epsilon \right) \quad (14)$$

$$\frac{\partial \omega}{\partial t} + \vec{V} \cdot \nabla \omega = C_{\omega 1} \frac{\omega}{\kappa} : \nabla \vec{V} - C_{\omega} \omega^2 + \nabla \cdot \left(\left(\frac{\mu_T}{\rho \sigma_\omega} \nabla \omega \right) \right) + (1 - F) \frac{2 \nabla \omega \nabla \kappa}{\sigma_{\omega, 2} \omega} \quad (15)$$

This implement $\kappa\epsilon$ equations in the wake region, where it has proven to be reliable. Whereas in the near wall region, where ϵ equation leads to a singularity, the model shifts towards the $\kappa\omega$. Hence although based on three balances equations, only two are implemented at the same time. The resolution of the two leads to the determination of the necessary turbulence quantities:

- μ_T : **turbulent viscosity**;
- l_T : **turbulent length scale**;

So the benefit of this model stays in the fact that, at wall, it is able both to solve balances and to employ wall functions to retrieve the necessary quantities. The choice between these two approaches is driven by the value of y^+ namely the non-dimensional distance from the wall. It is calculated as the ratio between the actual wall distance (y) and the viscous lengthscale (δ_v) that is in turn function of the molecular viscosity (μ), density(ρ) and frictional velocity (U_τ)

$$y^+ = \frac{y}{\delta_v}; \quad \delta_v = \frac{\mu}{\rho U_\tau}; \quad U_\tau = \sqrt{\frac{\tau_w}{\rho}} \quad (16)$$

Now, having y^+ equal or smaller than one, means that our grid permits to capture the necessary turbulence length scale, thus the model is able to correctly solve balances. Otherwise, in presence of greater values, it will use wall functions.

3 Preprocessing

The first part of the report describe path followed to set up the numerical model. Hence the definition of the domain and its discretization in finite volumes, followed by the definition of the boundary conditions and the numerical schemes.

3.1 Domain Definition

The whole domain has been divided into three different regions:

- Scroll
- Rotor
- Discharge

3.1.1 Scroll

The CAD model of this part of the domain was given directly by the Genova University then, in order to generate the so called *waterproof* geometry, the file was processed with CAD softwares with the aim of keeping only the internal volume.

3.1.2 Rotor

For what concern this region, things were a little bit more complicate, no CAD file was available.

However, thanks to some pictures and physical measurements, we were able to obtain a good correspondance. The model has then been used as a cutting tool for the interal volume thus directly reproducing the fluid domain.

3.1.3 Discharge

This part of the domain was generated starting from physical measurement performed directly on the test bench geometry setup. We could see that after the the original discharge section a flange is present that converges into a pipe where the thermodynamic quantities are measured.

Here a simplification is made: since in all the campaigns the waste gate valve is kept fully closed, we have chosen to not reproduce the complex geometry in correspondance of the WG discharge in order to simplify the already hard task of the mesher.

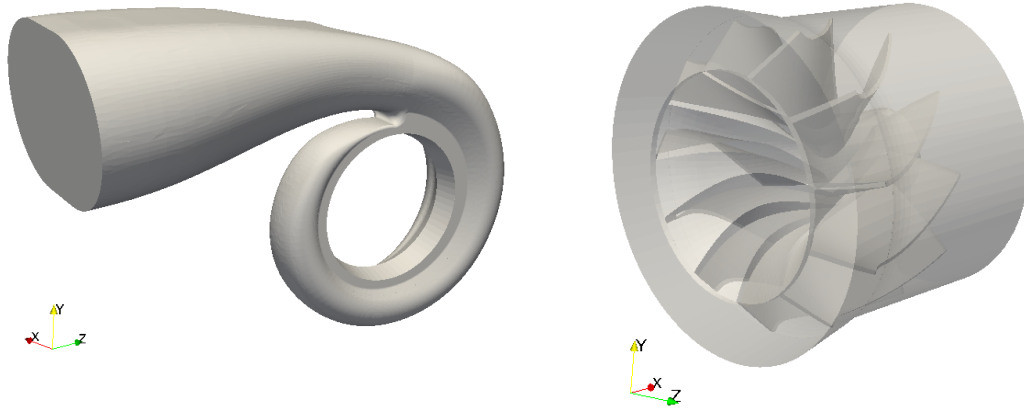


Figure 10: Volute and Rotor CAD files

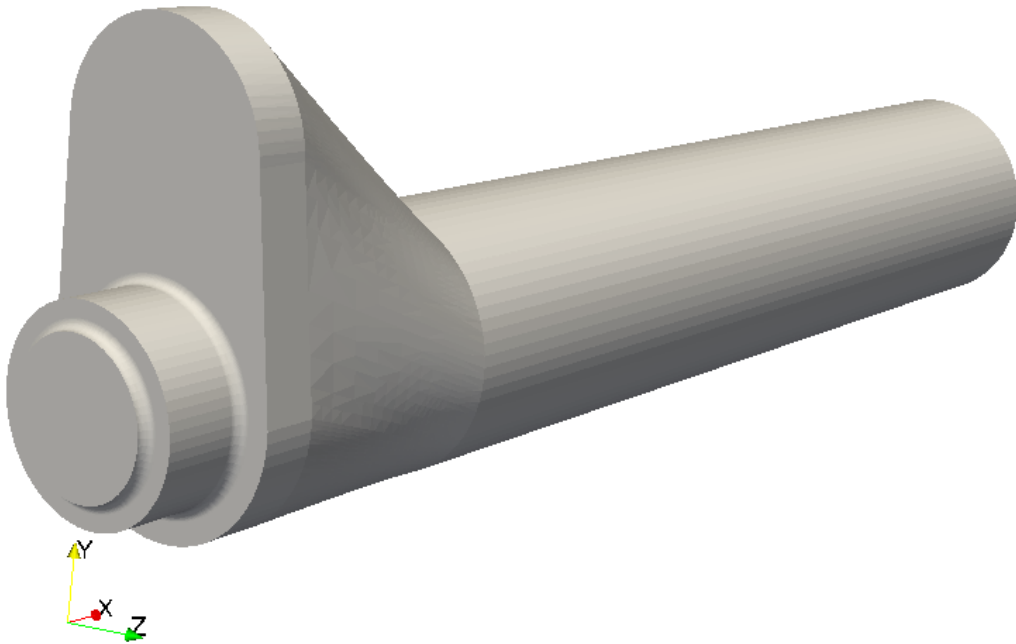


Figure 11: Discharge CAD file

3.1.4 Honeycomb Decoupler

As described before, the kernel of this work will be to investigate if, with suitable flow conditioner, we could create an even flow in order to be able to measure the efficiency of the machine only through simple boundary measurement. The first device we're going to model is the one that has provided the best experimental results in terms of pressure loss and de-swirling effect. It is composed by 2 cat-like honeycomb matrices inserted into an ad-hoc built expansion chamber. Thankfully all the technical drawings needed to reproduce the geometry were available so the CAD model was built straightforward.

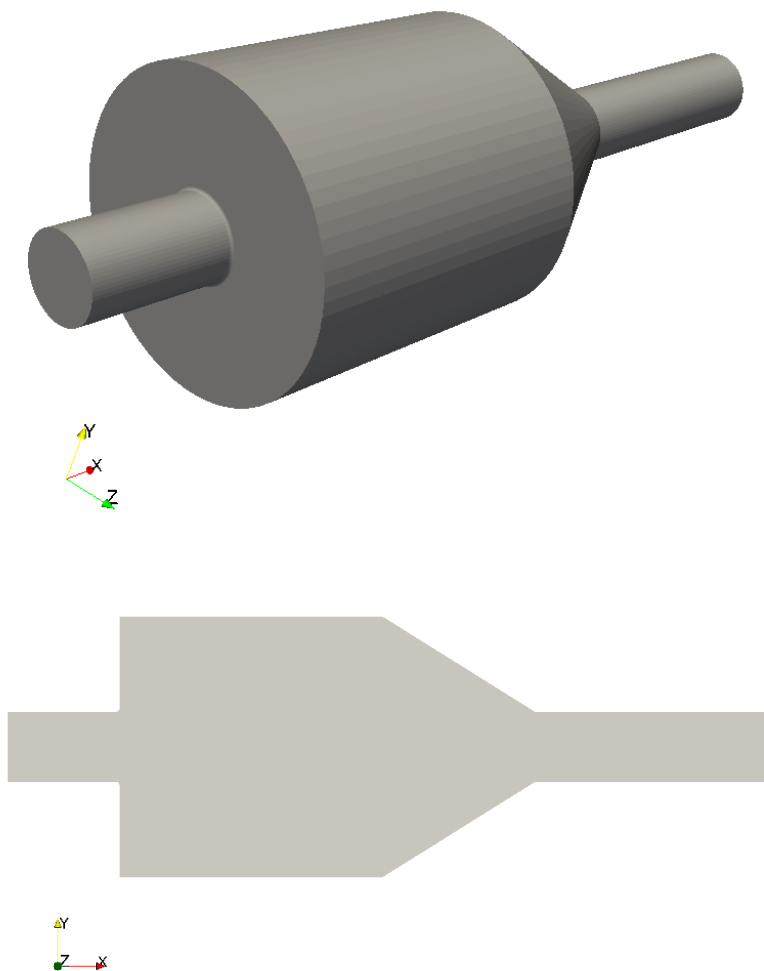


Figure 12: Honeycomb decoupler CAD file and section

3.2 Meshing strategy

Meshing is one of the most significant parts of CFD. The quality of the mesh can have a crucial impact on the accuracy of the results, or even determine whether a solution can be found or not.

We have decided to keep independent each region meshing, this leads to an accurate parameters setting and to a faster optimization process.

The best results were obtained exploiting `cfMesh` opensource library.

3.2.1 `cfMesh`

`cfMesh` is a cross-platform library for mesh generation [10] that is built on `OpenFOAM`. It is constructed by using components from the main library and it utilizes the concept of mesh modifiers, which allows efficient parallelisation. In addition, special care has been taken on memory usage, which is kept low by implementing data containers that do not require many dynamic memory allocation operations during the process.

`cartesianMesh` workflow has been utilised for meshing our geometry; it starts with the so-called mesh template from the input geometry and the user-specified settings. The template is later adjusted to match the input geometry. A predominantly hexahedral mesh is constructed, with polyhedral elements in the transition region between the cells of different size.

3.2.2 `cartesianMesh`

As for all the other `OpenFOAM` applications, the settings are specified through a specific dictionary file, in this case called `meshDict4`.

The mandatory settings, that have always to be specified, are the reference surface file and the maximum cell size. The `.stl` file is firstly transformed into an `.fms` file through the `surfaceFeatureEdges` utility. `.fms` surface file has the peculiarity to contain information both on the points and on the feature edges, thus it doesn't require an additional file to accurately reproduce sharp and sudden shape variations.

The next step is to exploit the patches already defined into the surface file or external files to specify local refinements in terms cell size. As we could see in 13 we have used this feature in order to have a very good discretization of the sharp edges mainly in the blades region.

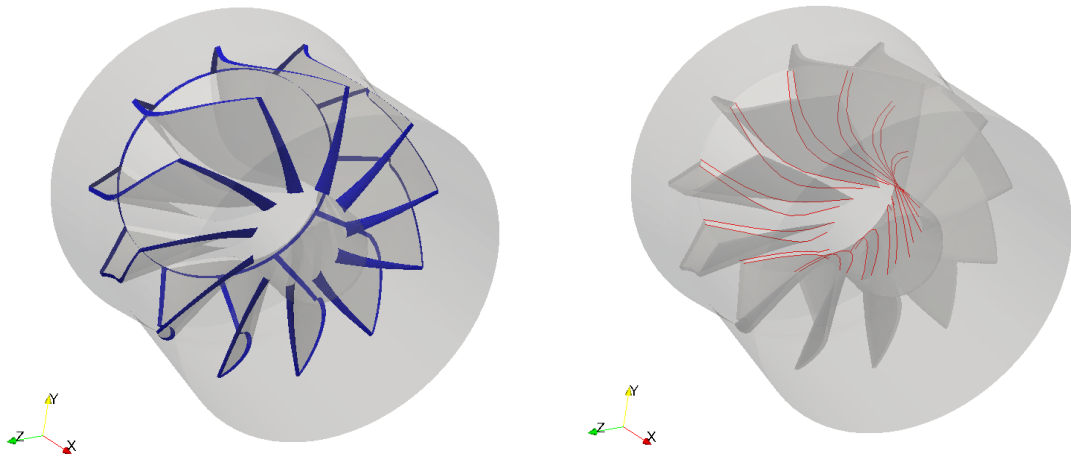


Figure 13: surface and edges files used to define further refinement

3.2.3 Meshing Result

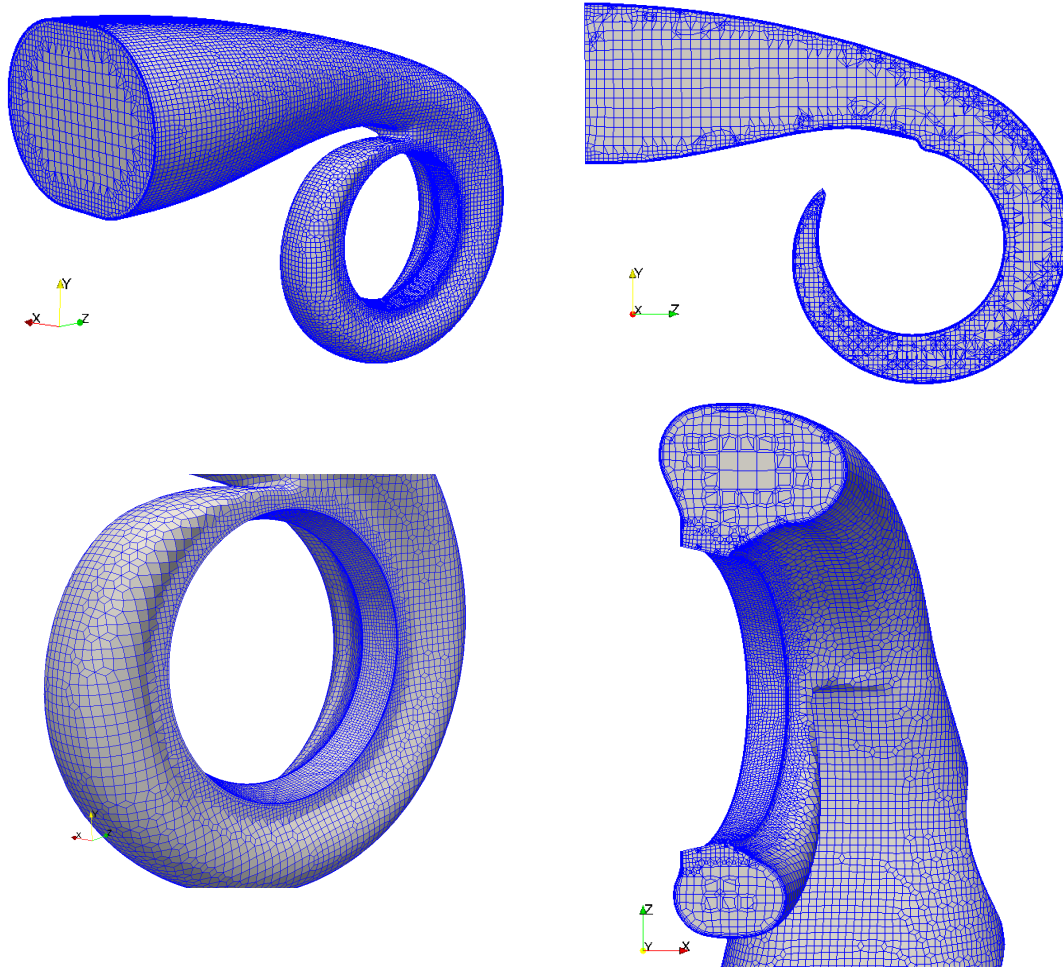


Figure 14: Volute Mesh result

Base cell	Inlet	Wall	Discharge
2 mm	2 mm	1.2 mm	0.5 mm

Table 2: Volute mesh sizing (element's edge length)

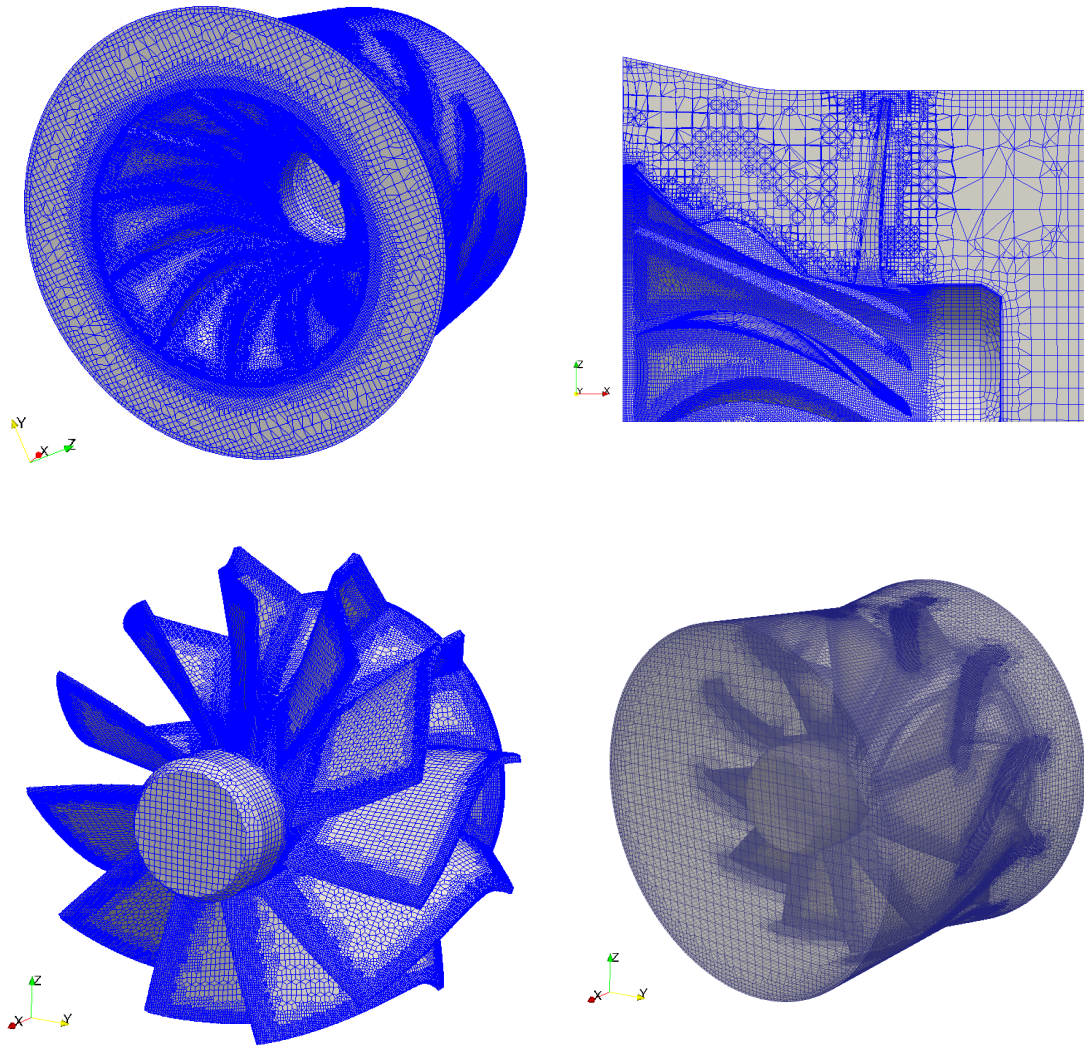


Figure 15: Rotor Mesh result

Base cell	Walls	Rotor Edges	Discharge
1 mm	0.5 mm	0.15 mm	0.5 mm

Table 3: Rotor mesh sizing (element's edge length)

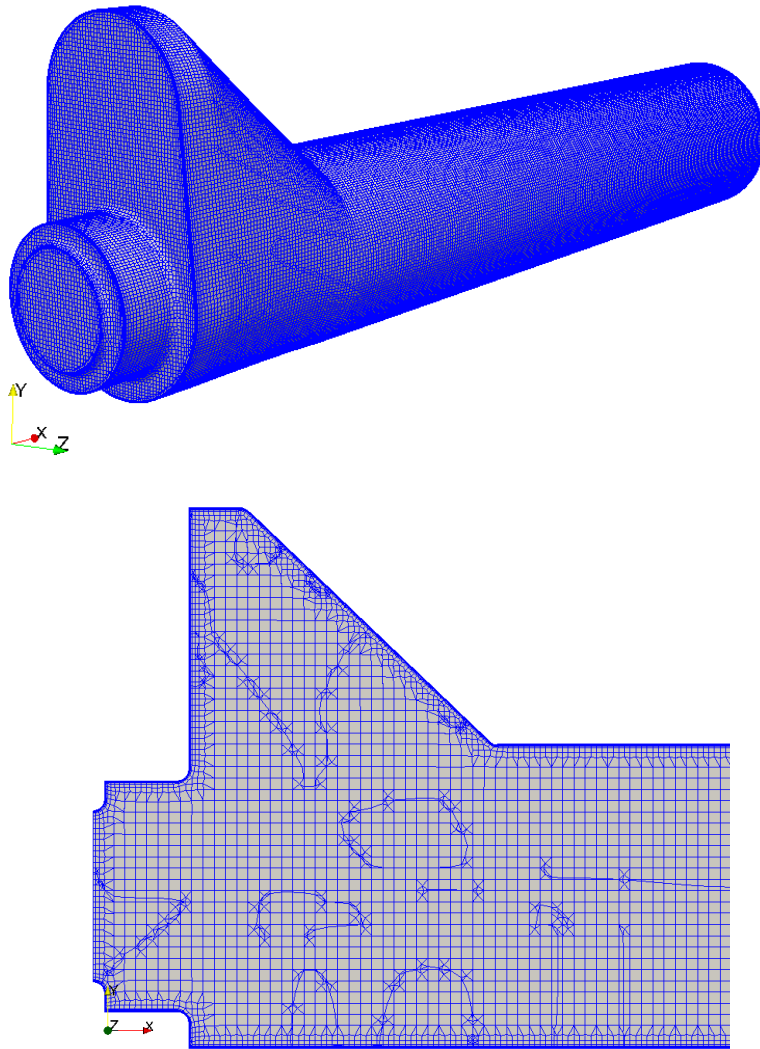


Figure 16: Discharge Mesh result

Base cell	Inlet	Wall	Outlet
2 mm	1 mm	2 mm	2 mm

Table 4: Discharge mesh sizing (element's edge length)

3.2.4 Rotating zone definition

Since we're simulating a rotating object, we need now to identify the cell zone that will actually move. This latter region is obviously the rotor one.

In order to distinguish it the `OpenFOAM` applications `topoSet5` and `setsToZone` are used. With the first we are telling the software to create a cell set called rotor that contains all the cells that will be displaced during the simulation, whereas with the latter we're simply converting a set to a zone.

3.2.5 Porous zone definition

We will see later on that the most efficient way to model the honeycomb matrix is to insert a porous source inside our model. As already done for the rotating region, also here we need to distinguish the cells where the dissipative source will be activated. The procedure is actually identical to the one depicted before⁶.

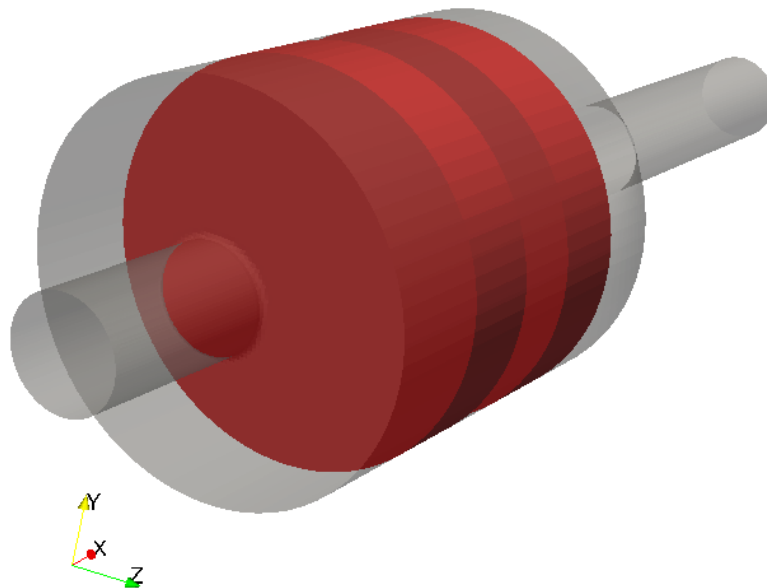


Figure 17: Porous cell zones

3.2.6 Mesh merging

Once all three regions are successfully meshed, we need to merge all into one single domain. This comes through the command `mergeMeshes`, that creates a single file with the infos coming from the three domains.

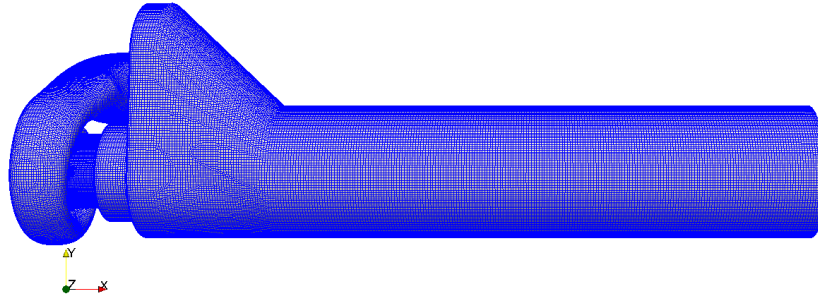


Figure 18: Merged Domain

The result is 1'357'111 total elements but, at this point, we still need to specify how to merge the interface regions.

This comes through **Arbitrary Mesh Interface** or **AMI**. This method permits to couple patches even though their elements do not coincide perfectly. As a matter of facts it performs at each time step an interpolation to successfully map the fields from master patch to slave one. The procedure is quite simple, it consists in changing the patches type and correctly defining the relationship between coupled faces.

It is now important to check the so-called AMI-weights⁸, values close to one mean that there's a good correspondence between the coupled faces, thus a good interpolation will be performed.

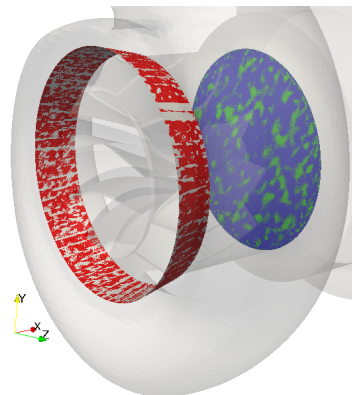


Figure 19: AMI coupled patches

3.3 Boundary condition

A problem is well posed if a solution exists and is unique; in order to achieve well posedness it is necessary to consider flow regime. For a compressible flow in 3D, 4 boundary conditions are needed: 3 conditions at inlet and one at outlet are needed.

Usually, two different sets of BC's could be used for turbomachinery applications:

- **Inlet** total pressure, total temperature, velocity direction. **Outlet** static pressure;
- **Inlet** mass flow-rate, total temperature, zero gradient static pressure. **Outlet** static pressure.

Referring to the experimental campaign we have at our disposal all the measurement needed to implement both set of initial conditions. So it's just a matter of finding the most stable and convergent one.

We are going to test both set of BC's in order to find out the one that permits to achieve the most accurate results.

The implementation dictionaries could all be found in the appendix B.1.

3.3.1 Temperature

As stated before at the inlet the **total temperature** along with the specific heat ratios is imposed. In the physical test bench, walls are insulated through fiberglass, so as a first trial the will be treated as **adiabatic**. Note that it is a strong assumption, as a matter of facts the result with this approach won't be satisfying, thus we will pass also to a **fixed temperature** condition (with values coming from experimental values) or an imposed **wall heat flux**.

3.3.2 Pressure

The inlet condition depends on the set of BC's we want to implement, in the case of total to static pressure ratio, obviously, **total pressure** is imposed. Whereas in the mass flow rate set, no fixed value is imposed. For what concern outlet section, instead of using a fixed value, in order to give some flexibility, it is preferable a **fixed mean value**. Since the boundary layer is iso-baric, **zero gradient condition** is imposed at walls.

3.3.3 Velocity

As for the pressure, we find different conditions depending on the BC set. In case of total to static pressure ratio, the velocity comes **through pressure** specified at the patch and it's direction is assumed normal with respect to the boundary. Otherwise, if **mass flow rate is imposed**, pressure will be then computed. To successfully simulate the BL development **adherence** condition is given at walls.

3.3.4 Turbulence

In order to deal with a turbulent flow we need to add another two boundary conditions. For what concern the **turbulent kinetic energy balance** it is possible to specify the **turbulence intensity** I defined as the ratio between the root mean square of the turbulence **velocity fluctuations** u' and the flow **mean velocity** U . From experience it could be state that 0.05 it's a reasonable value [17].

Shifting toward the **turbulent specific frequency** ω we need to specify th **turbulent mixing length**, it is a physical quantity describing the size of the large energy-containing eddies in a turbulent flow. A reasonable value for this kind of problem is quantified as the 20% of the inlet **hydraulic diameter**.

Although $\kappa\omega$ -SST is able to solve the balances at the wall, due to the complexity of the geometry and the high number of elements it was impossible to create boundary layer cells with a `firstLayerHeight` able to guarantee the non dimensional distance from the wall `yPlus` smaller than one allover the domain. Thus the solver will for sure employ `wallFunctions`. They permit to reproduce the velocity profile up to a certain distance from the wall without having to solve the balances. The specification of this features is again inside the turbulence boundary conditions.

3.4 Rotor Rotational Speed

The experiments have been focused on measurements at various n_{rid} , namely the reduced rotational speed, expressed as the ratio between the actual speed and the inlet total temperature $[rpm/\sqrt{K}]$.

3.4.1 Multiple Reference Frame Approach

Since we're modeling a rotating machine we need somehow to prescribe the law of motion of the rotor cell zone. The first and simplest approach comes through the MRF method. It applies to steady-state calculation since it does not involve any kind of mesh motion. The simulation of the rotating component comes through the addition of a term related to the *Coriolis* force and with the flux computed from the relative velocity and not the absolute one. The equations related to this approach will be explained later on. We need to add an proper dictionary to prescribe all the entries needed for this approach¹⁵

3.5 Porous Zone Set-up

As previously depicted, the most efficient approach to model an honeycomb matrix, comes through the application of a porous zone.

`OpenFOAM`, in order to deal with such a problem, inserts both an attenuation term γ in the time derivatives and a **sink-source** term S_i in the balances.

3.5.1 Darcy-Forchheimer model

In [11] it is reported the modeling of the sink coefficient according to *Darcy-Forchheimer* [11] model applied to a simple and homogeneous media. $D_{i,j}$ and F are the two parameters that drives the pressure loss, with the first taking into account the viscous dissipation and the latter the inertial one thus providing a loss curve function both of the velocity magnitude and its second power ($\Delta p = f(U, U^2)$).

$$S_i = -(\mu D_{i,j} + \frac{1}{2}\rho|u_{jj}|F)u_i \quad (17)$$

3.5.2 Coefficient set-up

The problem now stays in finding the right value for these coefficients. We have at our disposal the experimental measurements coming from the flow bench, that provide the pressure loss as a function of the mass flow-rate for the whole decoupler assembly depicted in 3.1.4.

The first step is now to build an ad-hoc model of the decoupler in order to numerically reproduce the flow-bench experiment. Domain and spatial discretization remain the same as the ones depicted in the previous sections.

In order to be consistent with the experimental test, the case is set up using inlet mass flow rate and outlet static pressure boundary conditions B.2. Walls are here treated as adiabatic. Now to tune the coefficient, the most effective way, is to vary the viscous one to obtain a good match with low flow rates, where the inertial dissipation is almost null. Subsequently, increasing the flow velocity, the Forchheimer term is tuned to obtain the "traditional" parabola curve.

3.5.3 Turbulence treatment

After having correctly defined the viscous and inertial influence of the media on the flow field, in order to be accurate, we need to evaluate also its influence on turbulence. To do that a standalone simulation of the matrix channels has been carried out.

Since it is unfeasible to model the entire domain with the necessary accuracy, periodicity has been exploited(see figs.2021).

Grid generation is here fundamental. Since we have to evaluate what happens to the two reference quantities k and ω , to be as much precise as possible, we have to make sure that turbulence balances will be solve and not modeled. As previously stated the necessary condition is to have the non-dimensional distance from the wall (y^+) smaller than unity. Moreover, since we're facing very low flow velocity (8-10 m/s), we could adopt the incompressibility option, thus energy equation won't be included in the resolution.

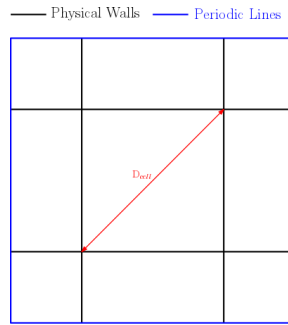


Figure 20: 2D sketch of the domain used for turbulence analysis

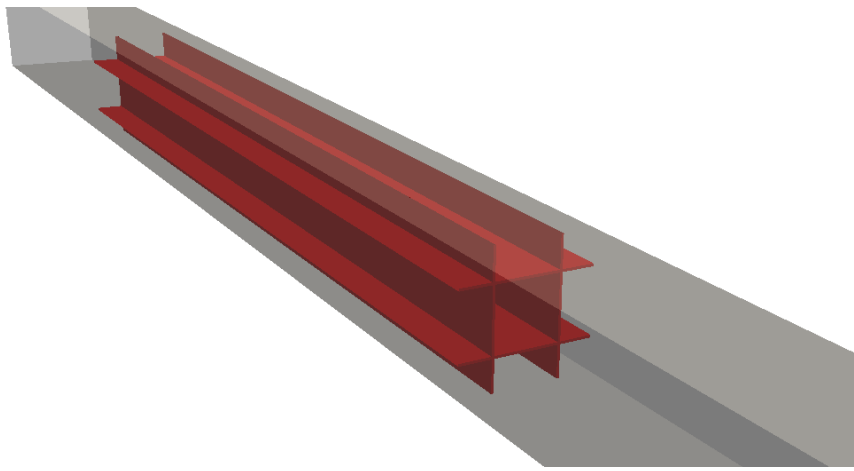


Figure 21: 3D representation of the domain used for turbulence analysis

Once imposed suitable boundary condition B.3, the aim is to evaluate what happens downstream of the channels in terms of kinetic energy and specific frequency. Finally we will have all the parameters that permits to avoid inserting the honeycomb geometry in the complete domain but, at the same time, having enough accuracy not to compromise final results.

3.6 Evaluating decoupler performances

To evaluate whether the conditioner is actually able to create an homogeneous flow field, an uniformity index γ_m referred to the measurement section has been introduced. The evaluation of this quantity starts from the computation of mass-flow rate average of the named field Φ . Being \dot{m}_i and A_i respectively the mass-flow-rate and the surface associated to each cell face i , the averaging procedure could be written as:

$$\bar{\Phi}_m = \frac{\sum(\Phi_i \dot{m}_i A_i)}{\sum(\dot{m}_i A_i)} \quad (18)$$

Subsequently the uniformity index:

$$\gamma_m = 1 - \frac{\sum [(|\Phi_i - \bar{\Phi}_m|)(\dot{m}_i A_i)]}{2\bar{\Phi}_m \sum(\dot{m}_i A_i)} \quad (19)$$

A value close to the unity associated with relatively small total pressure losses means a good device has been found. The computation of this index has been carried out using the axial velocity U_x as named field Φ because it is the variable that best describes the homogeneity at the discharge section.

However, we have to remember, we are studying geometries that must lead to accurate efficiency evaluation thus total temperature must be also take into account. In order to be consistent with experiments, we have chosen to evaluate this performance aspect with a random sampling process:

- Compute the efficiency through a mass-flow-rate averaging of the total temperature (enthalpy);
- Compute the efficiency through 1,2 or 3 random sampling points (arithmetically averaged);
- Check the entity of discrepancy between the two values.

Repeating this cycle for a sufficient number of time will produce a scatter plot of efficiency estimations. If the results are narrowly concentrated around the mass average value we could state that the device does its job correctly.

4 Solution Technique

The reference solver used for these kinds of simulations is `rhoSimpleFoam`.

4.1 The SIMPLE algorithm

SIMPLE is the acronym for **S**emi-**I**mplicit-**M**ethod-for-**P**ressure-**L**inked-**E**quations.

The SIMPLE algorithm was developed by Prof. Brian Spalding and his student Suhas Patankar at Imperial College, London in the early 1970s. Since then it has been extensively used to solve different kinds of fluid flow and heat transfer problems.

SIMPLE has been developed to deal with incompressible flow problems and its original structure could be found in several references [16]. In our case, the employed algorithm, has also to include the solution of the energy equation, in addition to continuity and momentum balances. The biggest difference between compressible and incompressible algorithms stays in the fact that, in the first case continuity is only a matter of momentum, whereas in the latter we have also the density effect. Thus the pressure correction equation isn't sufficient, we have to introduce also a step of density correction inside the solution process.

Since pressure appears in all three momentum equations and the velocity field has also to satisfy continuity, we do have four equations for four variables, hence the problem is closed. We are in front of a so-called pressure-velocity coupling, that permits to derive an equation for pressure starting from momentum and mass balances.

4.1.1 The Pressure Poisson Equation

The pressure Poisson equation [16] can be written starting from the momentum balance 20.

$$a_p \vec{U}_p = H(\vec{U}) - \nabla p \iff \vec{U}_p = \frac{H(\vec{U})}{a_p} - \frac{\nabla p}{a_p} \quad (20)$$

where

$$H(\vec{U}) = -\Sigma a_n \vec{U}_n + \frac{\vec{U}^o}{\Delta t}$$

The first term of $H(\vec{U})$ represents the matrix coefficients of the neighbouring cells multiplied by their velocities, while the second part contains the unsteady terms and all the sources except the pressure gradient.

The continuity equation could be discretised as 21

$$\nabla \cdot \vec{U} = \Sigma \vec{S} \cdot \vec{U}_f = 0 \quad (21)$$

where \vec{S} is the outward pointing face normal vector and \vec{U}_f the velocity on the face. This latter value could be obtained starting from 20 as follows:

$$\vec{U}_f = \left(\frac{H(\vec{U})}{a_p} \right)_f - \frac{(\nabla p)_f}{(a_p)_f} \quad (22)$$

At this point we can substitute this relationship (22) in the discretised continuity equation 21 to obtain the **Poission Pressure Equation**:

$$\nabla \cdot \left(\frac{1}{a_p} \nabla p \right) = \nabla \cdot \left(\frac{H(\vec{U})}{a_p} \right) = \Sigma \vec{S} \left(\frac{H(\vec{U})}{a_p} \right)_f \quad (23)$$

4.1.2 Algorithm Steps

In each iteration the solver performs the following operations in order to obtain all the necessary fields

- Solve the discretised momentum equation, starting from a guessed pressure field, to compute a temporary velocity field (momentum predictor);
- Compute the mass fluxes at the cells faces;
- Solve the pressure equation and under-relax the solution to obtain the new pressure field (pressure correction);
- Correct the mass fluxes at the cell faces;
- Correct the velocities on the basis of the new pressure field;
- Compute density from continuity equation;
- Compute Temperature solving energy equation;
- Correct density through the equation of state;
- Correct pressure field to guarantee continuity with the new density;
- Repeat till convergence.

The discretized momentum equation and pressure correction equation are solved implicitly, whereas the velocity correction is solved explicitly this is way it is called semi-implicit.

4.1.3 MRF equations

As briefly stated before, multiple reference frame approach, permits to treat rotating components in a steady state computation [12].

To start, we will firstly look how to express acceleration term for a rotating ($\vec{\Omega}$) frame.

$$\left[\frac{\partial \vec{r}}{\partial t} \right]_I = \left[\frac{\partial \vec{r}}{\partial t} \right]_R + \vec{\Omega} \times \vec{r} \Rightarrow \vec{u}_I = \vec{u}_R + \vec{\Omega} \times \vec{r} \quad (24)$$

In 24 we have written the relationship between the velocity expressed in term of the inertial frame \vec{u}_I and the rotational frame one \vec{u}_R , starting from the time derivative of position vector \vec{r} and rotational speed $\vec{\Omega}$. Then we can pass at the accelerations through another time derivative:

$$\left[\frac{\partial \vec{u}_I}{\partial t} \right]_I = \left[\frac{\partial \vec{u}_R}{\partial t} \right]_R + \vec{\Omega} \times \vec{u}_I \quad (25)$$

The develop of 25 leads to:

$$\left[\frac{\partial \vec{u}_I}{\partial t} \right]_I = \left[\frac{\partial \vec{u}_R}{\partial t} \right]_R + \frac{\partial \vec{\Omega}}{\partial t} \times \vec{r} + 2\vec{\Omega} \times \vec{u}_R + \vec{\Omega} \times \vec{\Omega} \times \vec{r} \quad (26)$$

We could now write the Navier-Stokes equations in the inertial frame with absolute velocity, for the sake of simplicity we report the incompressible case with constant molecular viscosity 27:

$$\begin{cases} \frac{\partial \vec{u}_I}{\partial t} + \vec{u}_I \cdot \nabla \vec{u}_I = -\nabla(p/\rho) + \nu \nabla \cdot \nabla(\vec{u}_I) \\ \nabla \cdot \vec{u}_I = 0 \end{cases} \\ \begin{cases} \frac{\partial \vec{u}_I}{\partial t} + \nabla \cdot (\vec{u}_I \otimes \vec{u}_I) - (\nabla \cdot \vec{u}_I) \vec{u}_I = -\nabla(p/\rho) + \nu \nabla \cdot \nabla(\vec{u}_I) \\ \nabla \cdot \vec{u}_I = 0 \end{cases} \\ \begin{cases} \frac{\partial \vec{u}_I}{\partial t} + \nabla \cdot (\vec{u}_I \otimes \vec{u}_I) = -\nabla(p/\rho) + \nu \nabla \cdot \nabla(\vec{u}_I) \\ \nabla \cdot \vec{u}_I = 0 \end{cases} \quad (27)$$

Next step is to exploit the relationship between the inertial and rotational frames in 25 to write NS equations in the relative frame with relative velocity:

$$\begin{cases} \frac{\partial \vec{u}_R}{\partial t} + \frac{\partial \vec{\Omega}}{\partial t} \times \vec{r} + \nabla \cdot (\vec{u}_R \otimes \vec{u}_R) + 2\vec{\Omega} \times \vec{u}_R + \vec{\Omega} \times \vec{\Omega} \times \vec{r} = -\nabla(p/\rho) + \nu \nabla \cdot \nabla(\vec{u}_R) \\ \nabla \cdot \vec{u}_R = 0 \end{cases} \quad (28)$$

28 could be further developed so the convected velocity is the one in the inertial frame, in particular:

$$\nabla \cdot (\vec{u}_R \otimes \vec{u}_R) = \nabla \cdot (\vec{u}_R \otimes \vec{u}_I) - \vec{\Omega} \times \vec{u}_R$$

and:

$$\nabla \cdot (\vec{u}_R \otimes \vec{u}_R) + 2\vec{\Omega} \times \vec{u}_R + \vec{\Omega} \times \vec{\Omega} \times \vec{r} = \nabla \cdot (\vec{u}_R \otimes \vec{u}_I) + \vec{\Omega} \times \vec{u}_I$$

Leading to 29 that corresponds to NS equation written for the rotating frame with convected absolute velocity:

$$\begin{cases} \frac{\partial \vec{u}_R}{\partial t} + \frac{\partial \vec{\Omega}}{\partial t} \times \vec{r} + \nabla \cdot (\vec{u}_R \otimes \vec{u}_I) + \vec{\Omega} \times \vec{u}_I = -\nabla(p/\rho) + \nu \nabla \cdot \nabla(\vec{u}_I) \\ \nabla \cdot \vec{u}_I = 0 \end{cases} \quad (29)$$

4.1.4 MRFrhoSimpleFoam

`rhoSimpleFoam` solver treats steady state problems of turbulent compressible flow. However, after the first calculation was carried out, we found out that something was wrong in terms of energy balance.

Even though we were facing an expansion, the temperature somehow increased. After having checked that all the settings were correct, the only sensible conclusion was an error in the energy equation.

It is useful to recall the named balance 30. Here we report the enthalpy formulation that is no other than the sum of internal energy e and kinematic pressure p/ρ ($h = e + p/\rho$).

$$\frac{\partial \rho h}{\partial t} + \nabla \cdot (\rho \vec{U} h) + \frac{\partial \rho K}{\partial t} + \nabla \cdot (\rho \vec{U} K) - \frac{\partial p}{\partial t} = -\nabla \cdot q + \nabla(\tau \cdot \vec{U}) + \rho r + \rho g \vec{U} \quad (30)$$

Now, since we're facing a steady state solver, all time derivatives have to be neglected, moreover we're not considering thermal sources (q) nor gravitational potential (g) but we need to take into account mechanical sources because the turbine is actually extracting power from the fluid.

At this moment, we're going to analyze the standard implementation of energy equation, reported in 1.

Listing 1: Standard energy equation implementation

```
fvScalarMatrix EEqn
(
  fvm::div(phi, he)
  + (
    he.name() == "e"
    ? fvc::div(phi, volScalarField("Exp", 0.5*magSqr(U) + p/rho))
    : fvc::div(phi, volScalarField("K", 0.5*magSqr(U)))
  )
  - fvm::laplacian(turbulence->alphaEff(), he)
  ==
  fvOptions(rho, he)
);
```

The code, starts by creating the matrix that has to be solved. With `he` we refer to the thermal quantity which has to be computed. It could be both internal energy e or enthalpy h , in fact, depending on the user choice, a different term is implemented for the kinetic energy field `K` or `Exp`. The operator `a?b:c`, literally translates in : "if `a` is verified then

evaluate \mathbf{b} , otherwise evaluate \mathbf{c} ".

Subsequently, we add the contribute of turbulent heat flux (`alphaEff`). The result is equated to sources contribution and the system closed. The problem relies in the fact that kinetic energy is evaluated using the magnitude of absolute velocity, whereas in presence of rotational frame of reference it has to be based on the relative velocity. Now, knowing that absolute velocity could be written as the composition of relative and rotational ones ($\vec{U} = \vec{W} + \vec{U}$), we could act to take into account this aspect.

Below it is reported the modified implementation.

Listing 2: Modified energy equation implementation

```
URel == U;
MRF.makeRelative(URel);

fvScalarMatrix EEqn
(
    fvm::div(phi, he)
    + (
        he.name() == "e"
        ? fvc::div(phi, volScalarField("Ekp", 0.5*magSqr(U) + p/rho))
        : fvc::div(phi, volScalarField("K", 0.5*magSqr(U)))
        - fvc::div( p*(U-URel))
    )
    - fvm::laplacian(turbulence->alphaEff(), he)
    ==
    fvOptions(rho, he)
);
```

First of all a new field `URel` is created through the MRF make relative object. Afterwards, from the kinetic energy field computed through absolute velocity \vec{U} we subtract the contribute coming from the rotational potential $\vec{U} - \vec{W}$. This will make us able to account for the energy extracted from the domain by the turbine.

4.1.5 rhoPorousSimpleFoam

Based on the already described `rhoSimpleFoam`, it deals with steady-state, turbulent and compressible problems with porosity treatment. The porous zone is added in the momentum equation as a source term, and it is seen like an added resistance. The formulation depends on whether we chose and implicit or explicit porosity modeling. Generally speaking, the first is preferred since it is more robust, thus it permits to achieve better and faster convergence. In this case, the momentum equation isn't solved alone but it comes through the pressure and a certain number of velocity field correctors.

4.2 Numerical Schemes

Navier-Stokes equation are actually impossible to be solved analytically. The aim of CFD calculations is to implement numerical models able to accurately reproduce the flow behavior.

Hence the need of specify which kind of discretization and solution method the solver has to adopt. All these information are contained into `fvSchemes` and `fvSolution` dictionaries.

4.2.1 Finite Volumes schemes

The set of terms, for which numerical schemes must be specified, are:

- `timeScheme`: first and second time derivatives $\frac{\partial}{\partial t}, \frac{\partial^2}{\partial^2 t}$
- `gradSchemes`: gradient ∇
- `divSchemes`: divergence $\nabla \cdot$
- `laplacianSchemes`: Laplacian ∇^2
- `interpolationSchemes`: cell to face interpolations.
- `snGradSchemes`: component of gradient normal to a cell face.
- `wallDist`: distance to wall, required by $\kappa\omega$ SST model.

The `default` discretization used for **gradients** is the **Gauss linear**, where **Gauss** entry refers to the standard finite volume discretization of Gaussian integration which requires the interpolation of values from cell centers to face centers. The proper discretization scheme is given by the `linear` entry, meaning central differencing.

Divergence terms includes both advection and other terms that are diffusive in nature. The fact that the terms are fundamentally different means that the `default` is set to `none` and each single entry is specified singularly. The non advective components generally use the **Gauss** integration along with `linear` interpolation whereas, the treatment of advective terms is one of the most crucial point in order to guarantee accuracy. Even in this case the schemes are all based on **Gauss** integration. We have tried different kind of interpolation schemes, each of it presents different accuracy and stability feature.

In particular, all the first calculations were initialized with `upwind` scheme that doesn't actually interpolate but it uses values from the upstream elements. Hence it is the most stable among all methods but it will lead to first order accuracy. On the other end, the highest accuracy will come with `limitedLinear` method. It is basically a `linear` scheme, hence it guarantees second order accuracy, but, in regions of rapidly changing gradient, it shifts towards an `upwind` method. It requires in addition a coefficient which varies from 0 to 1, where 1 is the strongest limiting, tending to linear as the coefficient tends to 0.

Sadly, during our calculations we were never able to achieve a good convergence using second order schemes. The main reason stays in the fact that higher accuracy tends to solve better instability phenomena such as flow detachments. These instabilities could not ever be solved with a steady-state solver.

However, refining the grid will help us to reduce some of the artificial numerical viscosity introduced by upwind schemes. `vanLeer` is another limited scheme, with less strong limiting then `limitedLinear`.

A **surface normal gradient** is the component, normal to the face, of the gradient of values at the centers of the 2 cells that the face connects. The basis of the gradient calculation at a face is to subtract the value at the cell center on one side of the face from the value in the center on the other side and divide by the distance. Since normally the mesh isn't aligned with the Cartesian co-ordinate, an explicit non-orthogonal corrector, known as the **corrected** scheme, can be added in order to gain second order accuracy.

The **Laplacian schemes**, typically contain the diffusive terms. **Gauss** scheme is the only choice of discretization. Then it is required to specify both an interpolation scheme for the diffusion coefficient and a surface normal gradient scheme. In the majority of the cases, a linear scheme it's applied for the interpolation of diffusivity whereas, for what concern the gradient, the case uses the same array of `snGradSchemes`.

The **interpolation schemes** sub-dictionary contains terms that are interpolation of values from cell to face centers, *e.g.* the interpolation of velocity onto the face centers to calculate the flux ϕ . **Linear** schemes are used in almost every case.

4.2.2 Finite Volumes solutions

The equation solvers, tolerances and algorithms are controlled from the `fvSolution` dictionary. The `solvers` entry specifies each linear solver, method to solve a matrix equation, that is used for each discretized equation. Before discussing the various methods, it has to be noted that the matrices are sparse. Consequently, the solvers are generally iterative meaning that they are based on reducing the equation residual over subsequent solutions. The residual is evaluated by substituting the current solution into the linear system and taking the magnitude of the difference between left and right hand sides; it is then normalized in order to make it independent of the scale of the problem.

Before solving an equation, the residual is evaluated starting from the current values of the field, this is the so-called **initial residual** useful to check the convergence of the solution. After each solver iteration the residual is re-evaluated. The solver stops if any of the following conditions are reached:

- the residual falls below the solver tolerance, `tolerance`;
- the ratio of current to initial residuals falls below the solver relative tolerance, `relTol`;
- the no. of iterations exceed a maximum, `default = 1000`;

Mainly `relTol` is set to 0 to force the solution to converge at each time step, thus it permits greater accuracy but at a higher computational cost.

Talking about the solvers, we see that, for pressure field, `GAMG` is the choice. Geometric-algebraic multi-grid solvers uses the principle of generating a quick solution on a mesh with a smaller number of elements and then mapping it onto a finer mesh, using it as initial guess. `GAMG` is faster than standard methods when the increase in speed by solving first on coarser meshes outweighs the additional cost of refinement and mapping. The user is only required to specify an approximate mesh size at the most coarse level: `nCellsinCoarsestLevel`. The `smoother` entry, specifies which smooth solver it is employed to solve the equations at the various refinement levels. `GaussSeidel` is the preferred choice.

The other fields are all solved through `GaussSeidel smoothSolver`.

Another sub-dictionary present in `fvSolution` is `relaxationFactors`. It controls under-relaxation, a technique that permits to weight the results both on the current solution and on the previous one, *e.g.* $p = \alpha p^{new} + (1 - \alpha)p^{old}$ where $0 \leq \alpha \leq 1$. It is a technique useful to guarantee stability and convergence especially in steady-state solvers.

5 Standard duct

5.1 Turbine Map

The first thing we want to calculate is whether our model is able to correctly reproduce the real machine behavior. We have at our disposal experimental data regarding reduced mass flow (m_{rid}) as a function of the total to static expansion ratio (ϵ_{TS}). Moreover, thanks to the similitude theory, once the reduced rotational speed (n_{rid}) is fixed, the curve should be more or less equal for every value of inlet total temperature.

In 5 we could see the test conditions in which the computation has been carried out.

Initial Conditions		
n_{rid}	4750	rpm/\sqrt{K}
$T_{T_{in}}$	353.15	K
n	89263	rpm
p_{out}	102250	Pa
β_{TS}	1.35 .. 1.6	

Table 5: Turbine Map BC

At this point it is only a matter of run the simulations and ,for each pressure ratio, read the mass flow rate and compute the reduced one as in 31:

$$m_{red} = \dot{m} \frac{\sqrt{T_{T_{in}}}}{p} \left[\frac{kg \cdot K^{1/2}}{s \cdot bar} \right] \quad (31)$$

In 22 we could see the result obtained. First thing we note, is that the numerical model underestimate the mass flow treated by our machine. The main reason of this difference could be found in experimental measurement errors but also in the fact that rotor has been modeled starting from very approximate dimensions. However, the most important thing, is that the trend of the curves is pretty much equal. This means that the real machine and the modeled one behaves in the same way when shifting from one operating point to another.

Moreover, we have repeated the test increasing the inlet total temperature while maintaining the same reduced speed (thus increasing the rpm). The two curves are almost the same, highlighting the fact that similitude has been respected.

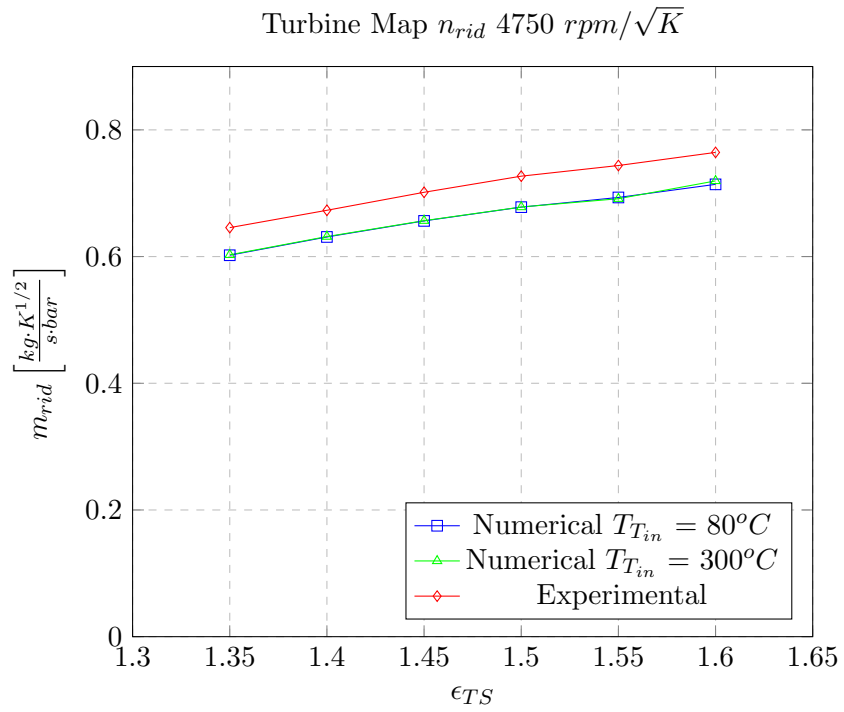


Figure 22: Turbine Map

5.2 Adiabatic Case

Next thing is to check the performances of our model from the energetic point of view. Keeping in mind that efficiency measures comes directly from temperature ones, we need to correctly set-up all the model parameters to obtain such an accuracy. Since no direct data are available in terms of heat losses, the best thing is to start from a completely adiabatic model. From these result it will be possible to notice the biggest differences with respect to the experimental case and in which direction we must act.

In the last paragraph we have seen that, once imposed the expansion ratio, the model gives back a mass flow rate slightly lower than the measured one, as a consequence, if the mass flow rate boundary conditions are imposed, we will obtain an higher expansion ratio with respect to experiments. Because the temperature (enthalpy) drop is strictly related to the pressure drop, we have chosen to fix β_{TS} and leave free \dot{m} .

We will now refer to the numerical result associated with:

- **Upwind** numerical schemes;
- **Cold-Flow** condition: $T_{T_{in}} 80 \text{ }^\circ\text{C}$;
- **Smallest** total to static expansion ratio: $\beta_{TS} 1.4$;

5.2.1 Mesh Statistics

Three different meshes are treated here:

- Without BL cells;
- BL cells only in discharge section;
- BL cells in the whole domain.

Since adding Boundary Layer Cells means also increasing the total number of cells and so the computational time, we want also to quantify the difference between different grids in order to check whether it will be possible to avoid this feature. In 6 the most significant mesh quality measures are reported.

Quality Index	No-BL	BL-disch	BL-tot
No of elements	999'771	1'357'111	3'165'526
Max Non-Ortho	67.08	67.06	67.76
Avg Non-Ortho	9.06	8.05	9.41
Max Skewness	4.225 (1)	4.219 (1)	5.997 (404)
Max Aspect Ratio	8.6	20.98	30.53

Table 6: Grid features, standard duct discharge

5.2.2 **yPlus**

As stated in the turbulence chapter, $\kappa\omega$ -SST model is able both to solve balances at the wall and to apply wall function to retrieve the necessary quantities. The choice between these two approaches is driven by the value of a quantity called **yPlus**, namely the non-dimensional wall distance. In 7 we see the average values of this quantity in the three different spatial discretization.

Patch	Avg y^+ No-BL	Avg y^+ BL-disch	Avg y^+ BL-tot
Volute	62	62	4
Rotor	15	15	0.9
Discharge	27	0.7	0.7

Table 7: Average y^+ values, standard duct discharge

5.2.3 Residuals

As we already pointed out, the convergence of the simulation, could be monitored looking at the initial residuals at each iteration. We could state we fall approximately below $1e-5$ (especially pressure wise) the computation could be considered as converged. These values mainly depends on:

- Grid Quality;
- Numerical Schemes;
- Solver Set-up;
- Robustness of the boundary conditions;

In 8 we could see the results of the adiabatic, upwind calculations.

Field	No-BL	BL-disch	BL-tot
Pressure	1e-6	3.7e-6	3.3e-6
Velocity X	8.3e-7	2.2e-6	2.5e-6
Velocity Y	9e-7	1.9e-6	1.1e-6
Velocity Z	6.5e-7	1.2e-6	1.7e-6
Enthalpy	1.4e-6	8.3e-7	9.5e-7
Turbulent Kinetic Energy	1.7e-7	5.4e-7	9.9e-8
Specific Frequency	4.5e-8	6.8e-8	6.2e-8

Table 8: Initial residual values, standard duct, adiabatic

5.2.4 Graphic Post Processing

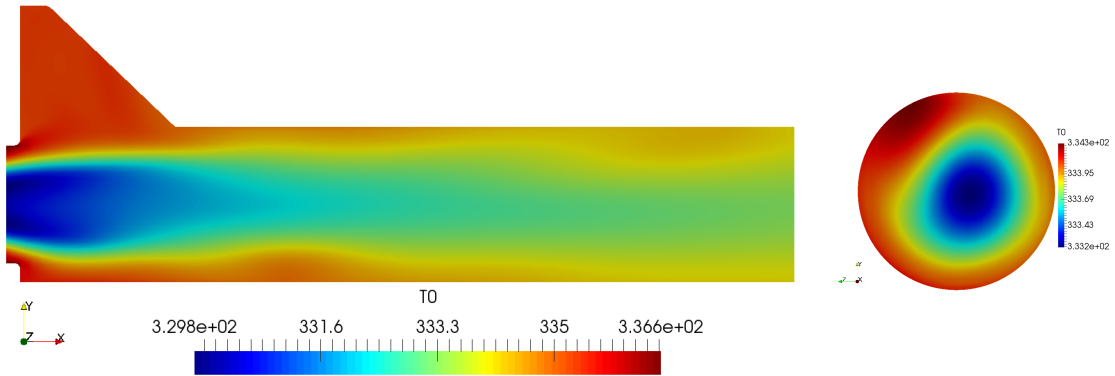


Figure 23: Total temperature contours, cross section and measurement plane, standard duct adiabatic

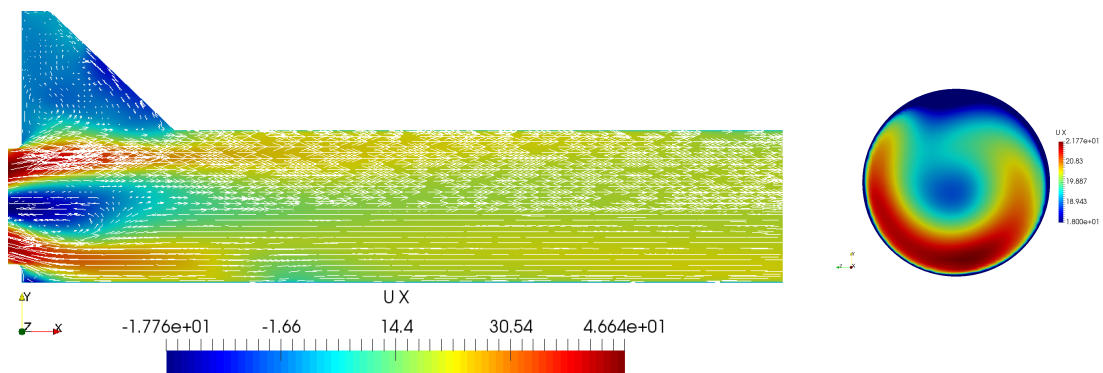


Figure 24: Axial velocity contours and vectors, cross section and measurement plane, standard duct adiabatic

5.2.5 Numerical Results

Now, in order to compare the results, we will extract data in some specific points through the `probes` function. As we could see in 25 the computation overestimates total temperature. Since the walls are treated as adiabatic, the simulation does not take into account effects of thermal losses, as a matter of facts, note that the temperature is raising when we move towards the boundaries. Data corresponds to a 0° sampling, so the probes travels aligned with the y-axis.

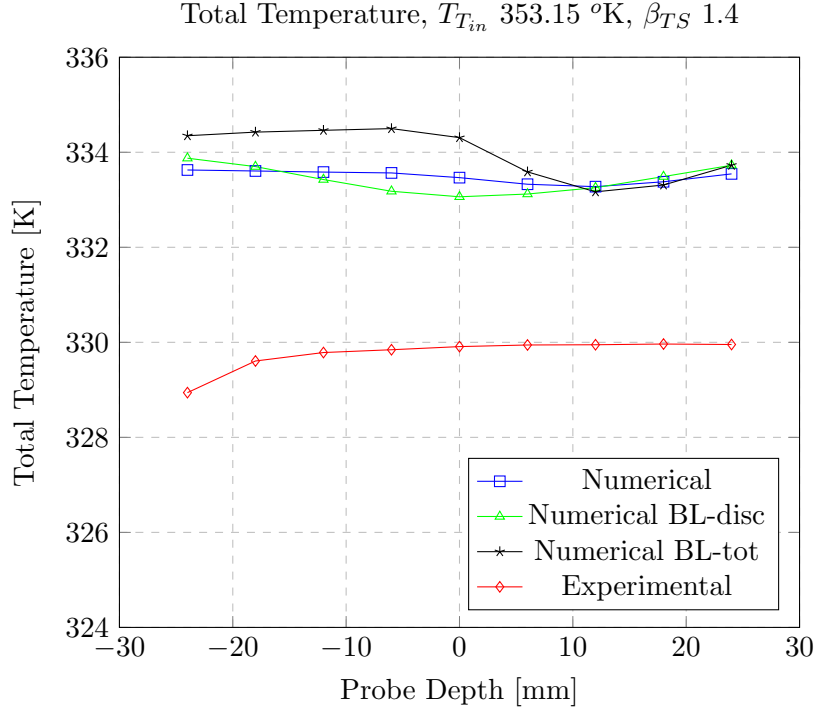


Figure 25: Total temperature, standard duct, adiabatic walls

For what concern the comparison of the grids, we see that in the bottom half of the measurement section, the results are quite consistent one with each other. The biggest disparities are in the top half. Sadly however, also the experimental in this region are affected by the heat exchange with the measurement device 4.

So now it is clear that we have to provide some heat exchange in order to close up with the actual experiment, but let's see the results also in term of axial velocity.

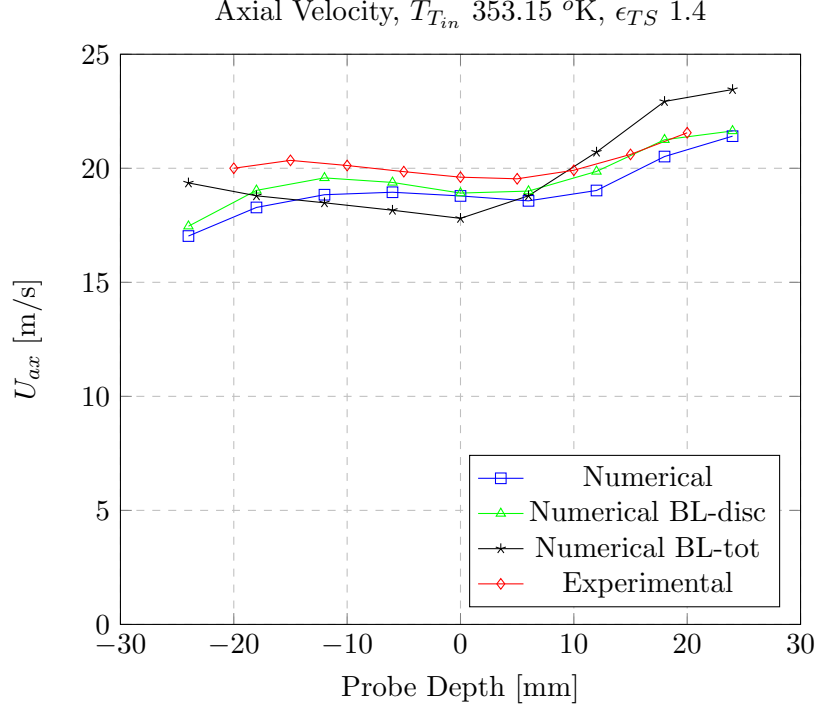


Figure 26: Axial velocity, standard duct, adiabatic walls

In this case we have already found a good correspondence of the values, we expected slightly lower numerical values because being the mass flow rate smaller, also the meridional velocity should be lower (assuming equal density and cross section).

In particular, the best results, are the ones coming from the grid with boundary layer cells only on the discharge section. Since it has also provided good numbers for the temperature field, it will be the one used hereafter.

5.2.6 Meridional and tangential velocity maps

Since we have at our disposal the contour representation of the axial velocity field constructed on the basis of the experimental data, in order to validate our model it could be useful to compare results. Even though there is a slight difference between the scales used in the 2 plots. we could appreciate the fact of a good matching between calculation and measurement.

For what concern swirl 28, we do not have any truthful experimental counterparts. In fact, since this field is calculated through pressure differential technique, facing low speeds means also pressure difference of few hundreds Pascals thus we're not able to distinguish actual measures from sensor's sensitivity.

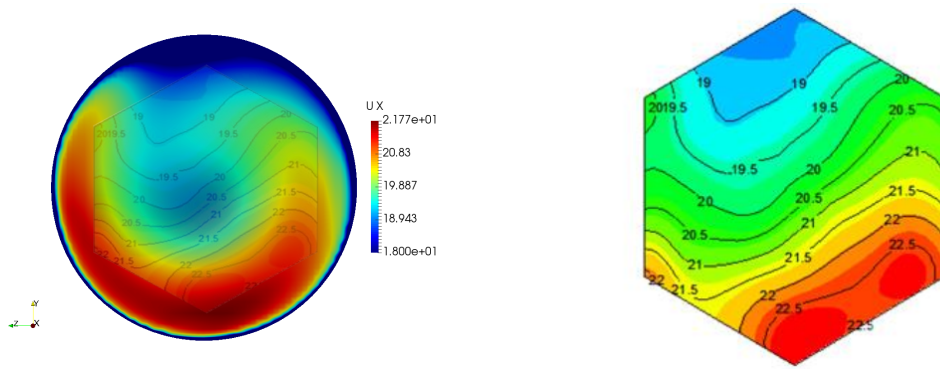


Figure 27: Axial velocity contours at measurement section, numerical (left) experimental(right)

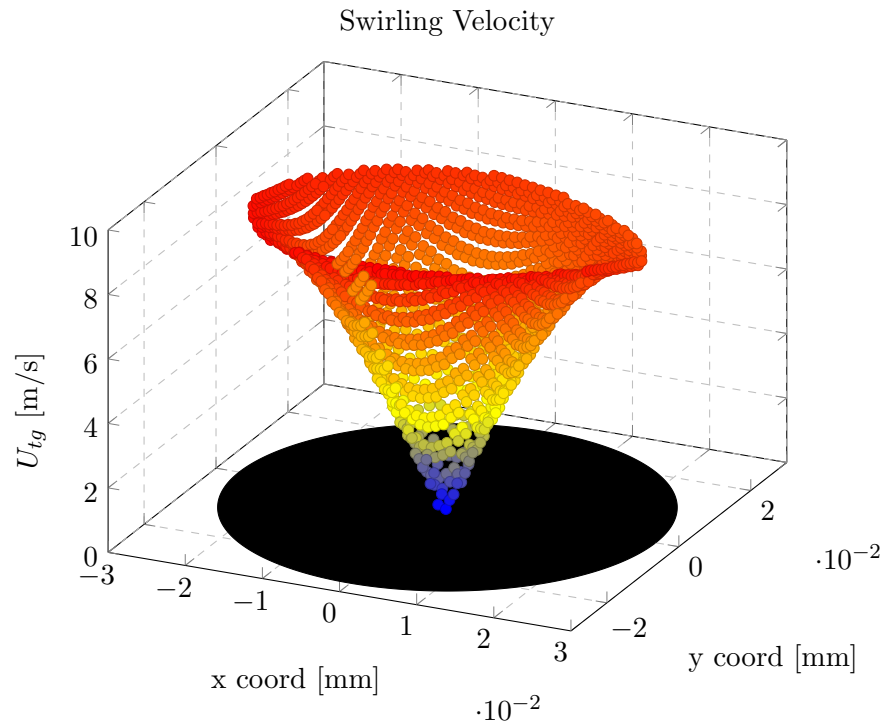


Figure 28: Tangential velocity, standard duct, adiabatic

5.3 Fixed Wall Temperature

Since for this set of experimental conditions we have available measurements of wall temperature at the measurement plane, as a first trial, we could try to impose it on the whole discharge portion, and see what changes from the energetic point of view. Keep in mind that all the volute and rotor are still treated as adiabatic so the changes will be modest. The value of imposed static temperature is $57^{\circ} C$ thus $330 K$. Moreover as previously depicted, the employed grid is the one with BL cells only on discharge.

5.3.1 Residuals

Pressure	1.2e-6
Velocity X	2.3e-7
Velocity Y	2.1e-7
Velocity Z	2.1e-7
Enthalpy	2.1e-6
Turbulent Kinetic Energy	2.8e-7
Specific Frequency	1.1e-7

Table 9: Initial residual values, standard duct, fixed wall temperature

5.3.2 Graphic Post Processing

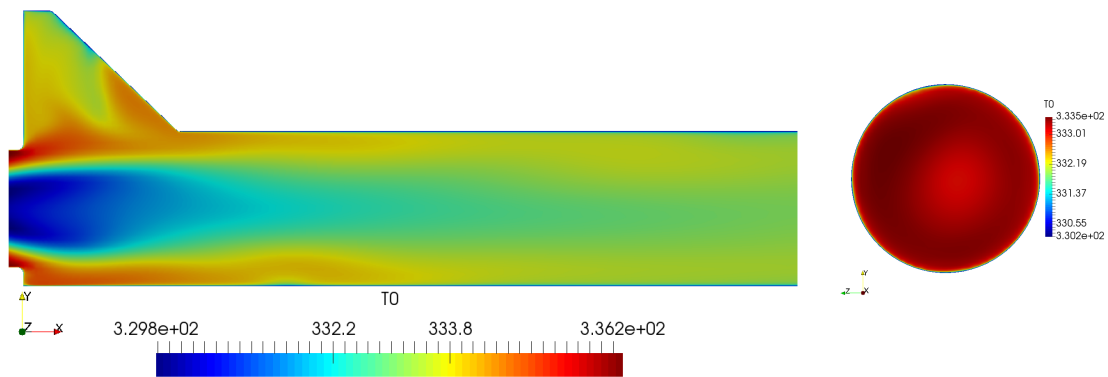


Figure 29: Total temperature contours, cross section and measurement plane, standard duct fixed wall temperature

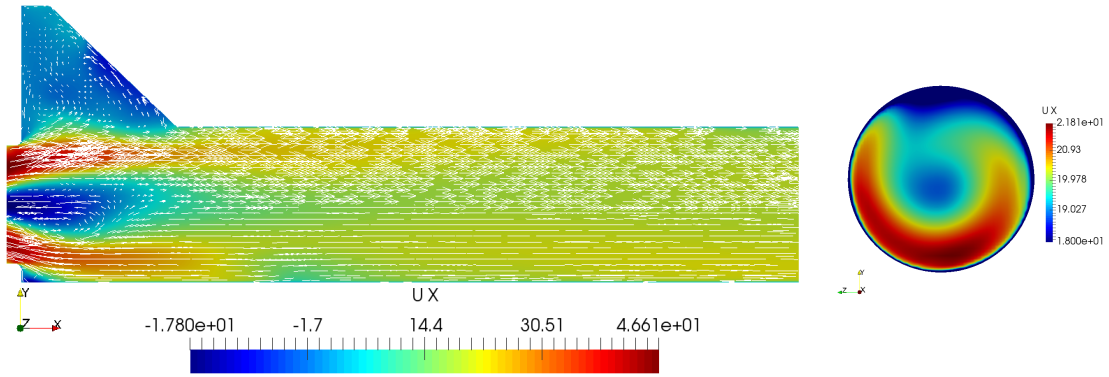


Figure 30: Axial velocity contours and vectors, cross section and measurement plane, standard duct fixed wall temperature

5.3.3 Numerical Results

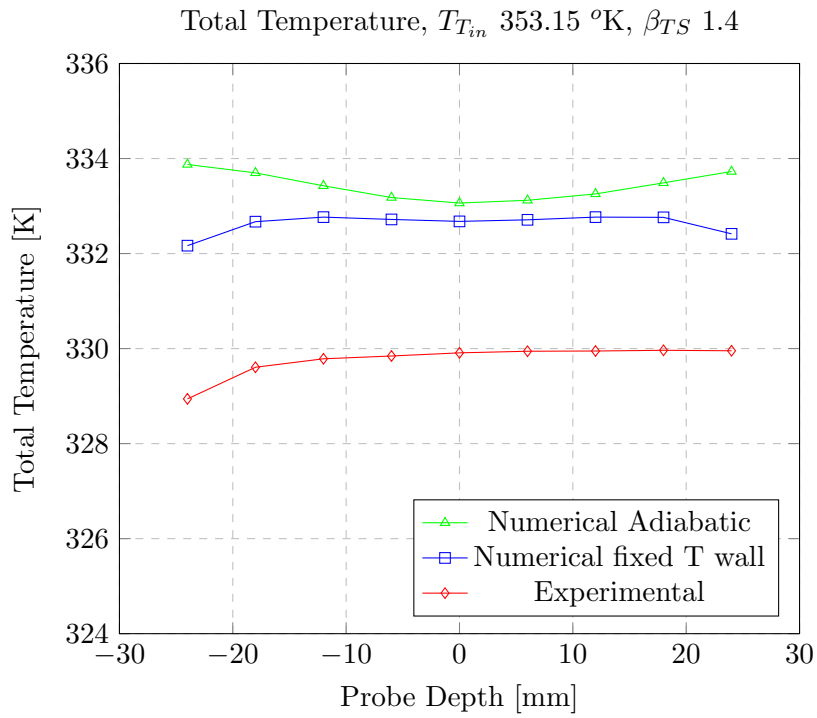


Figure 31: Total temperature, standard duct, fixed wall temperature 330 K

In 31 we could see that now, the trend of the total temperature pretty much corresponds to the experimental one but, the average value is still higher. That could stay in the fact that the hottest part of the process doesn't take into account any thermal loss. Please note that we have also to face the uncertainty of the measures.

From the momentum point of view 32, the addition of a fixed temperature constraint doesn't imply any substantial change. As a matter of fact the meridional velocity profile computed, still looks very similar to the physical one.

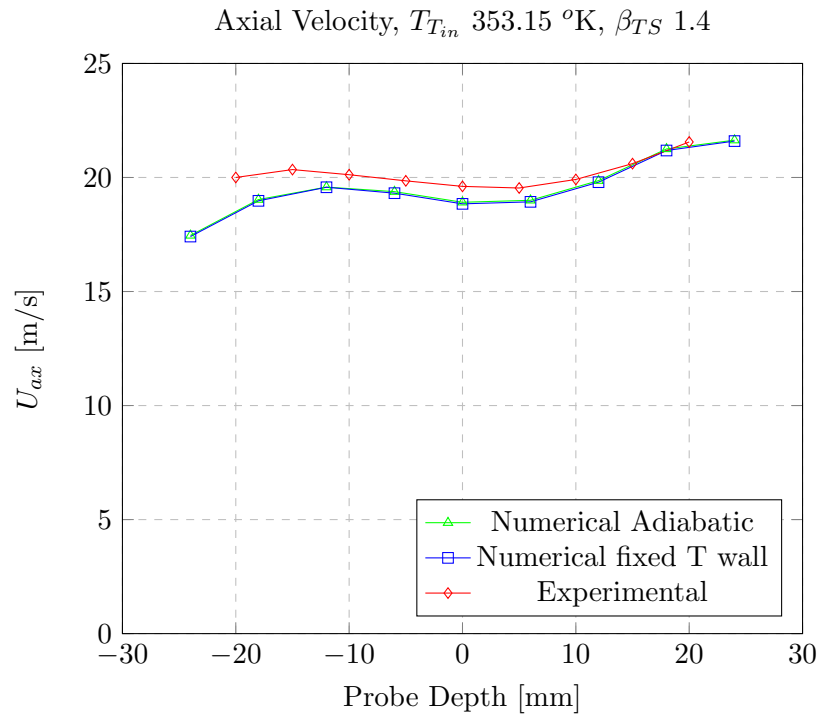


Figure 32: Axial velocity, standard duct, fixed wall temperature 330 K

5.4 Wall Heat Flux

From physical point of view, the most reasonable thing to simulate thermal losses, is imposing a wall heat flux. Sadly, however, we do not have any data to support us in the calculation of such a variable. It is function of, for example, Reynold's number, temperature difference with the ambient, thermal conductivity of the wall etc.

A simple approach, to obtain a trial value, consists in calculating the average temperature difference between the adiabatic numerical and real experimental data. This, multiplied times the specific heat at constant pressure and the mass flow rate, gives us a value of the thermal power to be exchanged with the ambient.

The solver, requires heat flux "q". This latter value could be obtained straightforward by dividing the total power times the total wall surface A_{ex} as briefly reported in 32.

$$Q = \dot{m}c_p\Delta T [W] \quad q = Q/A_{ex} [W/m^2] \quad (32)$$

The calculation led to a thermal flux of approximately $2950 [W/m^2]$. This value has been imposed uniformly on all walls of the domain even though, from experience, we're pretty confident to state that the region involving higher temperatures will for sure present an higher heat exchange rate. Nevertheless, at this point, we must accept this approximation since there isn't any other way around.

5.4.1 Residuals

Pressure	5.5e-6
Velocity X	3.5e-6
Velocity Y	2.8e-6
Velocity Z	1.5e-6
Enthalpy	1.2e-6
Turbulent Kinetic Energy	1.1e-6
Specific Frequency	1.1e-7

Table 10: Initial residual values, standard duct, wall heat flux

5.4.2 Graphic Post Processing

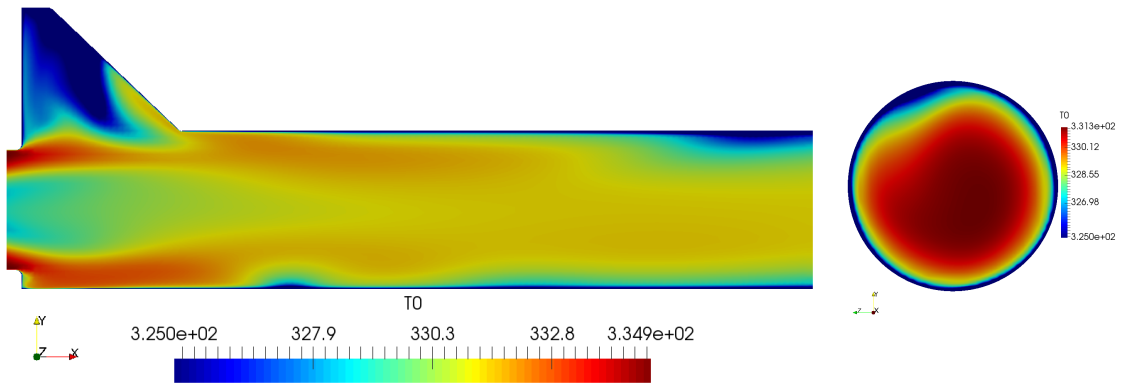


Figure 33: Total temperature contours, cross section and measurement plane, standard duct wall heat flux

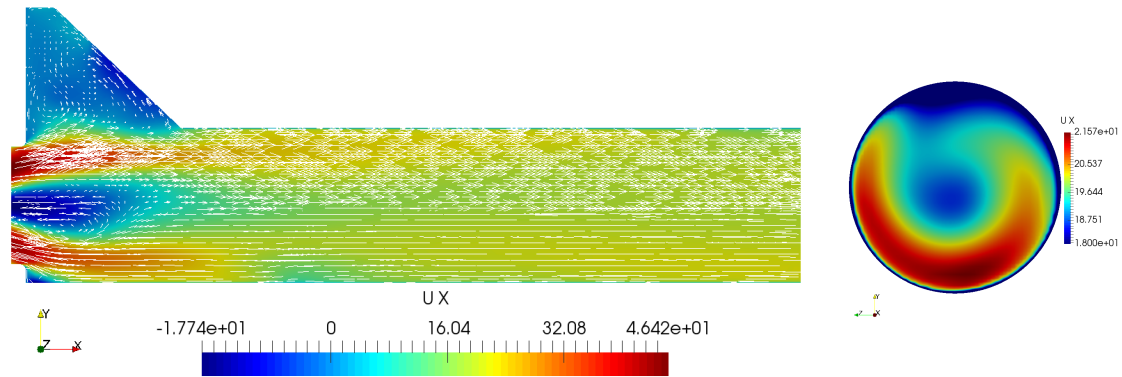


Figure 34: Axial velocity contours and vectors, cross section and measurement plane, standard duct wall heat flux

5.4.3 Numerical Results

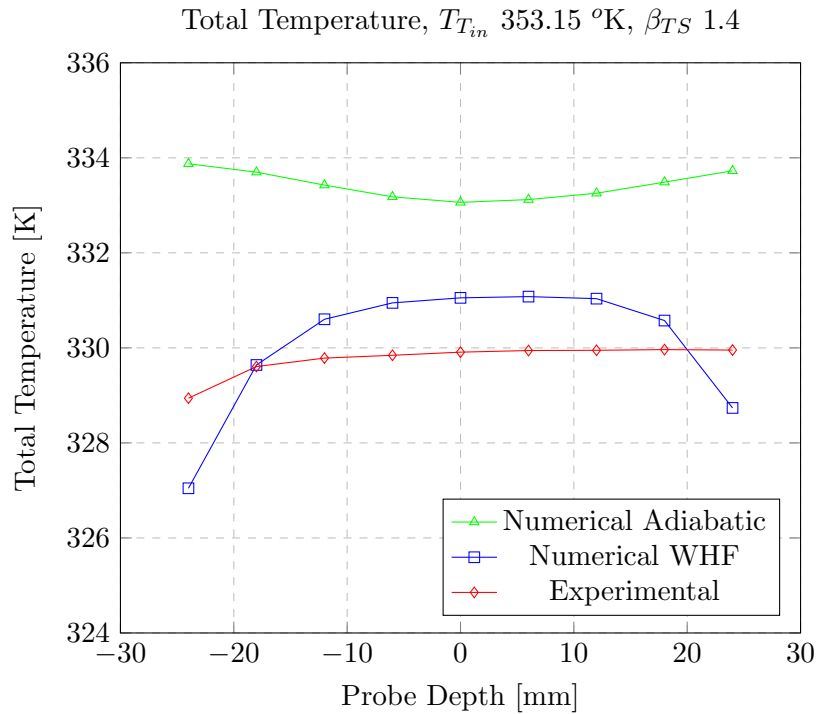


Figure 35: Total temperature, standard duct, wall heat flux 2950 W/m²

In 35 we see that, now, the average value of the total temperature has dropped to more or less 1 K above the experimental value. However, its trend along the radial coordinate, present a much higher concavity. This comes from the fact that the heat flux has been homogeneously distributed along the domain whereas, as already pointed out, downstream from the rotor, since we're facing lower temperatures, we would expect less heat losses. The conclusion is that we must think of a different way to spread the heat exchange.

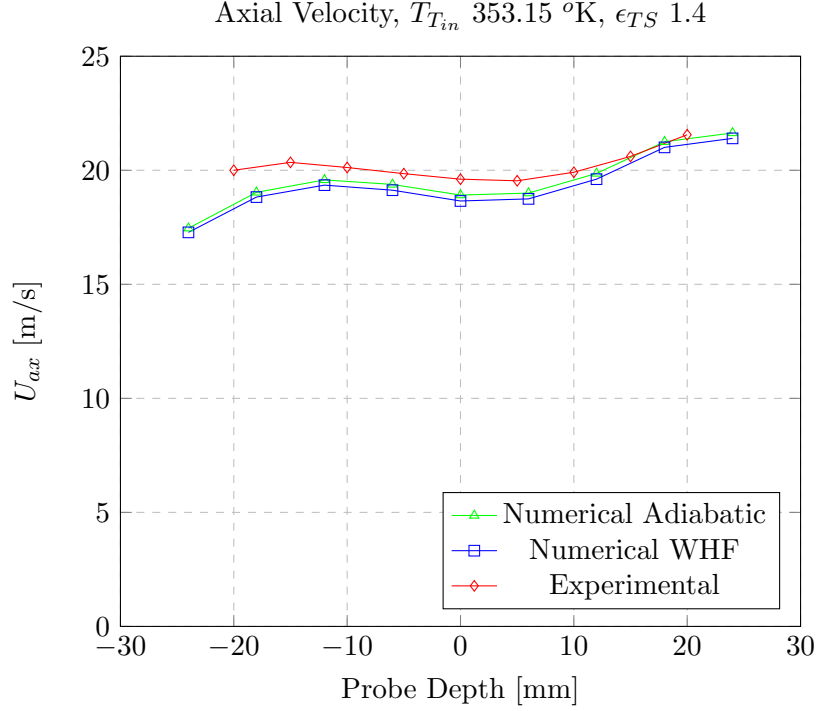


Figure 36: Axial velocity, standard duct, wall heat flux 2950 W/m²

5.5 Uneven Wall Heat Flux

Since we want to impose that the biggest part of the heat exchange takes place in the volute section, we have chosen to assign it 2/3 of the total heat power Q and the remaining third in the rest of the domain. This will help us to smooth the total temperature radial distribution hence leading towards the experimental results.

5.5.1 Residuals

Pressure	2.3e-6
Velocity X	1.6e-7
Velocity Y	1.2e-7
Velocity Z	8.9e-8
Enthalpy	2.4e-6
Turbulent Kinetic Energy	4.8e-7
Specific Frequency	9.8e-8

Table 11: initial residual values, standard duct, uneven wall heat flux

5.5.2 Graphic Post Processing

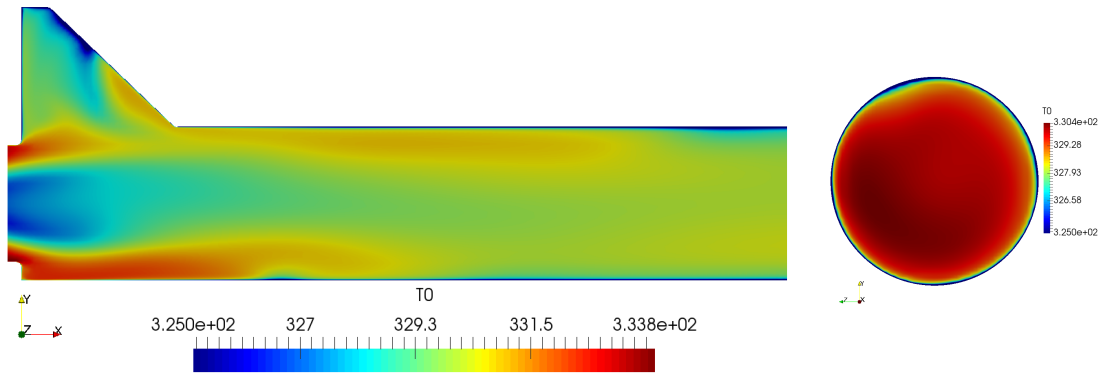


Figure 37: Total temperature contours, cross section and measurement plane, standard duct, uneven wall heat flux

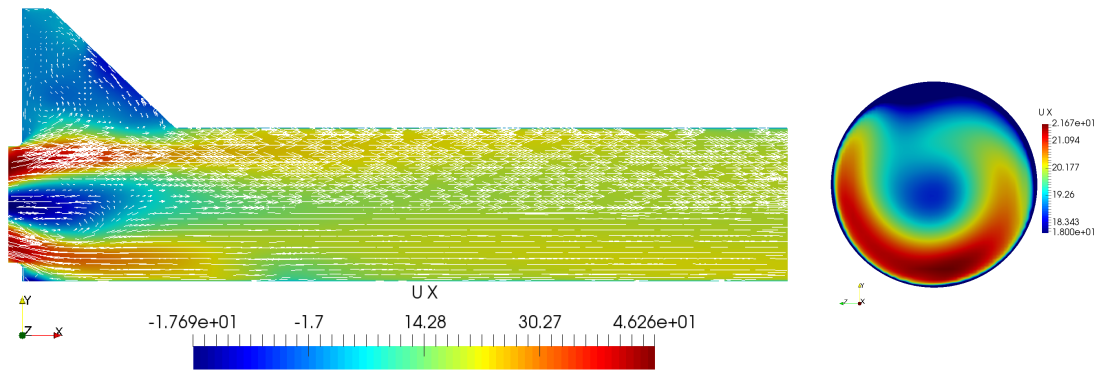


Figure 38: Axial velocity contours and vectors, cross section and measurement plane, standard duct, uneven wall heat flux

5.5.3 Numerical Results

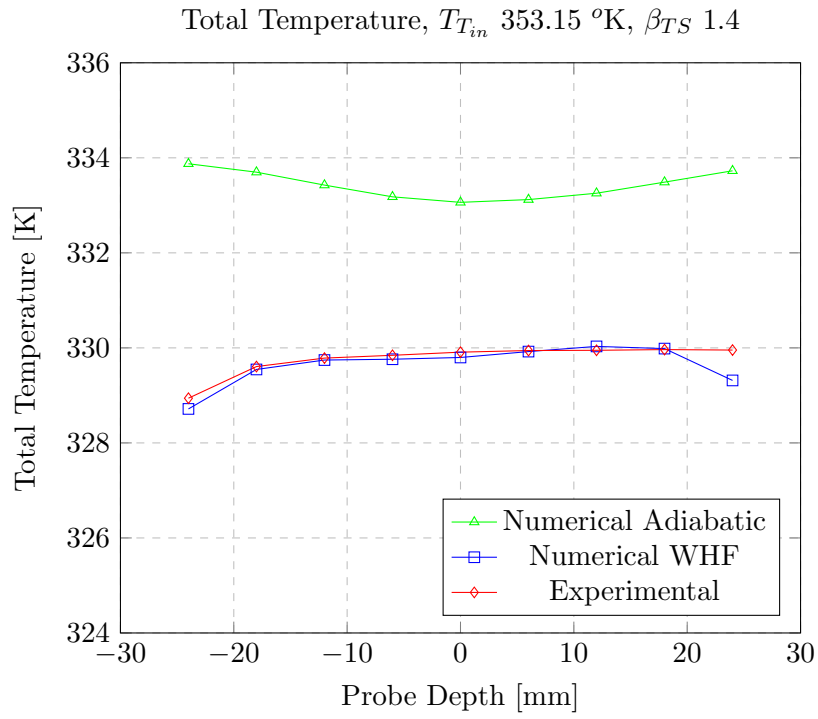


Figure 39: Total temperature, standard duct, uneven wall heat flux

Figure 39 shows that we have finally found out the correct numerical set-up of our model. It is now able to reproduce almost correctly the total temperature distribution. In 40, the meridional velocity seems almost equal to the ones obtained before, the trend looks like the experimental values whereas, in term of magnitude, it is slightly lower.

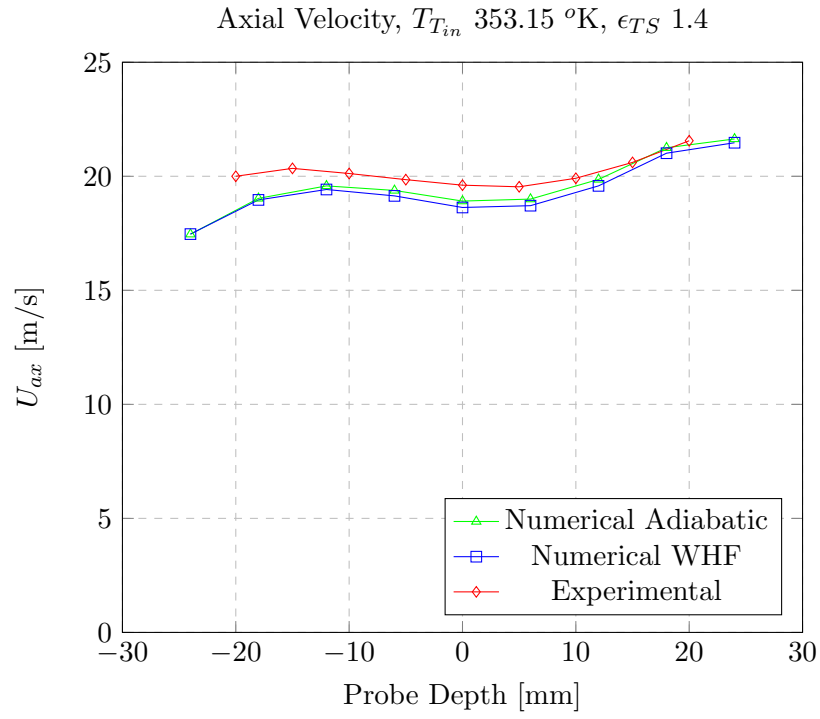


Figure 40: Axial velocity, standard duct, uneven wall heat flux

5.6 Grid sensitivity analysis

As in all computational fluid dynamic cases, we have to verify the so call grid independence. Thus a more refined spatial discretization has been created and then, if the quantities are within a certain range of base ones, we could state that results do not depend anymore on grid sizing. It's important during this tests to maintain constant the size of boundary layer's cells so to exclude the effect of possible different turbulence treatment (wall functions or not) on results.

Base cell	Inlet	Wall	Discharge
1 mm	1 mm	0.8 mm	0.5 mm

Table 12: Refined volute Mesh Sizing (element's edge length)

Base cell	Walls	Rotor Edges	Discharge
0.5 mm	0.5 mm	0.15 mm	0.5 mm

Table 13: Refined rotor mesh sizing (element's edge length)

Base cell	Inlet	Wall	Outlet
1 mm	1 mm	1 mm	1 mm

Table 14: Refined discharge mesh sizing (element's edge length)

The above tables reports the sizing adopted for the domain discretization, leading to 2'041'584 elements.

In the following fig.41, we could see that results for the 2 grids are quite similar. In particular:

- 2% Average difference on meridional velocity;
- 0.05% Average difference on total temperature.

So, hereafter, we deem that it is not necessary to start ad additional "refined grid" campaign to confirm previously stated results.

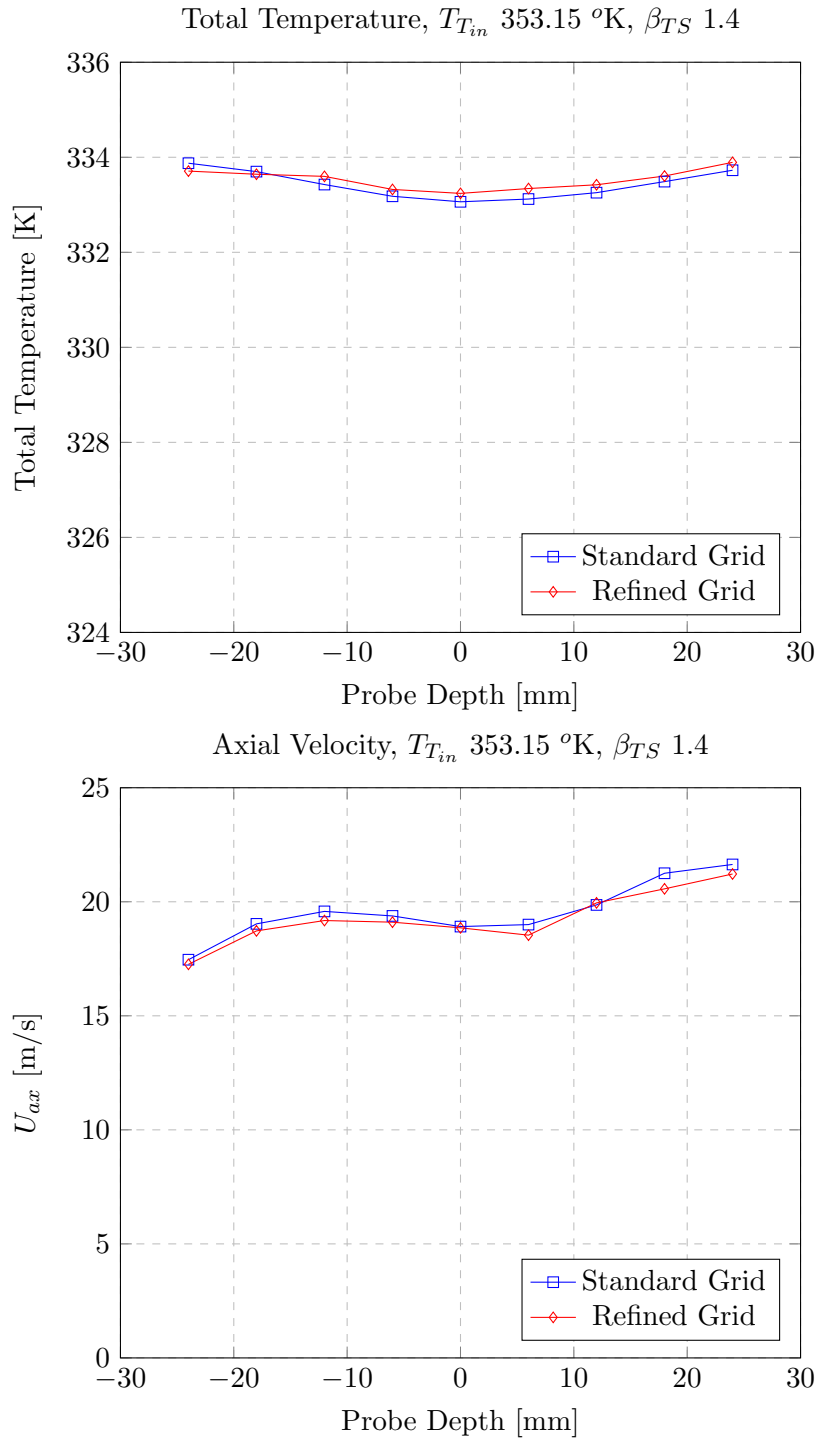


Figure 41: Total temperature and axial velocity, grid independence analysis standard duct

6 Honeycomb Expansion Chamber

6.1 Porous Coefficient Set Up

Here we report the results obtained with the procedure depicted in 3.5.2.

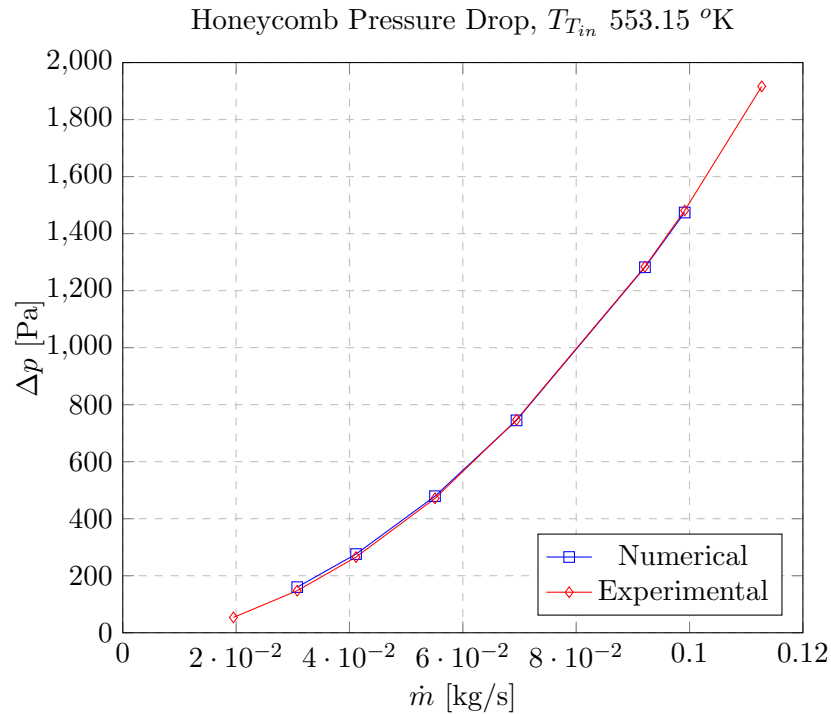


Figure 42: Experimental and numerical honeycomb expansion chamber flow meter

A good match has been obtained, the two curves are almost one over each other. The correspondent value of the viscous loss coefficient D is $2e6$ whereas the inertial one F has been found equal to 10.

In 15 we report as an example the value of the residuals for the case with the highest mass-flow rate, for the other ones equal or better values were achieved.

Pressure	8.9e-6
Enthalpy	1.3e-5
Turbulent Kinetic Energy	9.8e-7
Specific Frequency	2.7e-8

Table 15: Initial residual values, porous coefficient set-up

6.2 Turbulence Analysis

In this section we will show the results coming from the analysis of the honeycomb matrix effect on turbulence.

6.2.1 Meshing

No. cells	Base size	Wall size	BL layers
1'507'916	1 mm	0.25 mm	4

Table 16: Grid Features, honeycomb turbulence analysis

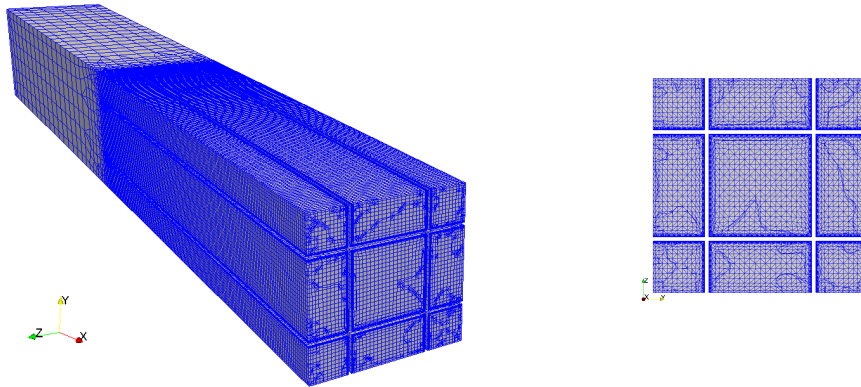


Figure 43: Honeycomb channel mesh and cross section detail

6.2.2 Results

The simulation was carried out for 2500 iterations, using second order discretization (`limitedLinearV 1`) for the velocity gradients.

\mathbf{p}	\mathbf{Ux}	\mathbf{Uy}	\mathbf{Uz}	κ	ω
$2.7e-6$	$3.5e-7$	$4.1e-6$	$4.1e-6$	$5.3e-6$	$5.8e-8$

Table 17: Initial residuals, honeycomb turbulence analysis

First check is on the y^+ that, as stated before, must be lower than one. With an average value of 0.0042 we completely undergo the constraint.

For what concern the turbulent kinetic energy (κ) and specific frequency (ω), results were acquired in two different sections: one just after the duct outflow and one $6D_{cell}$ downstream to see the effect of mixing.

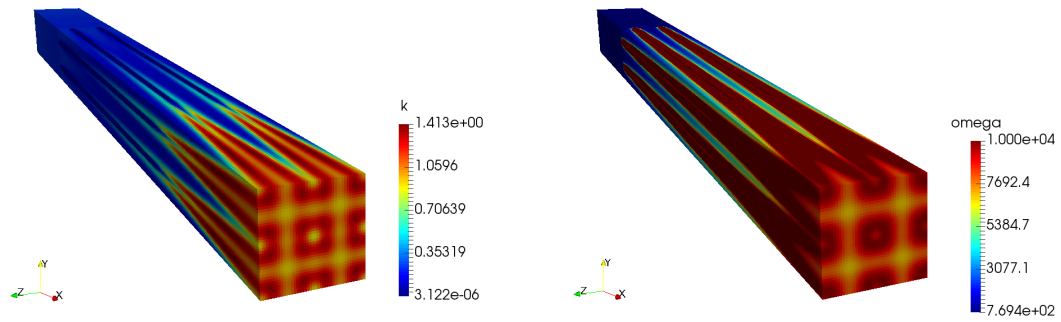


Figure 44: Turbulent kinetic energy and specific frequency, whole domain

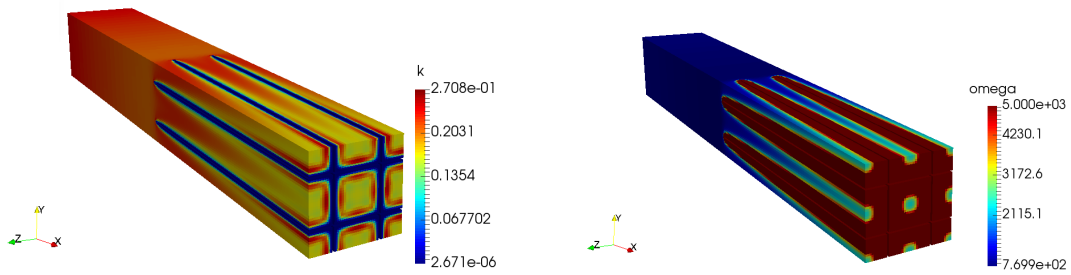


Figure 45: Turbulent kinetic energy and specific frequency, half channel

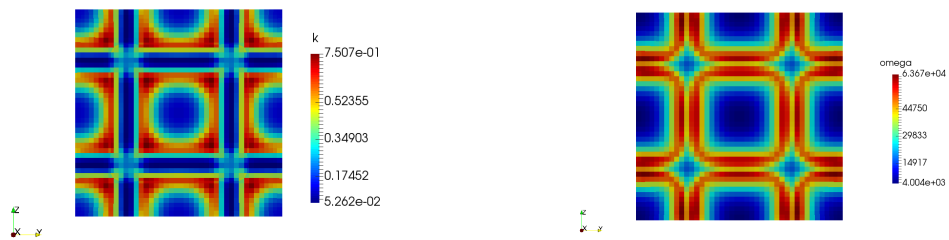


Figure 46: κ and ω contours, first measurement section



Figure 47: κ and ω contours, second measurement section

Inlet κ_{avg}	Sect.1 κ_{avg}	Sect.2 κ_{avg}	Inlet ω_{avg}	Sect.1 ω_{avg}	Sect.2 ω_{avg}
0.24	0.377	1.138	894	32'615	8'668

Table 18: κ and ω average values, turbulence analysis

From the graphical post processing in fig.44 to 47, it is possible to evaluate the trend of the two turbulence fields. In particular we see that turbulent kinetic energy is almost demolished once the flow has entered the honeycomb channel. Instead, looking at the specific frequency, the opposite phenomena happens. After the outflow, the effect of mixing is pretty clear: the kinetic energy tends once again to increase whereas the opposite for the frequency.

At this point in order to introduce a good modeling of this porous media we can act in the following way:

- Add a source term to the honeycomb cell zone that acts to reduce κ and increase ω .
- Define another cell zone downstream of the previous one and add a source to reproduce the effects of mixing.

The implementation of such a procedure is reported in appendix B.4.

6.3 Results: first operating point, adiabatic walls

The first operating point we're going to analyze is the one closest to the tests done in case of no flow decoupler:

Test Condition	MU	Value
n_{rid}	$rpm\sqrt{K}$	4750
$T_{T_{in}}$	K	353.15
\dot{m}	kg/s	0.05

The procedure is analogue to the previous numerical campaign, in particular focus has been taken on the heat losses. We're going to their effects on the total temperature as well as on the velocity. For this situation, we have found the total to static pressure ratio set of boundary condition struggled during the initialization of the fields leading to simulation crash. Hence mass-flow-rate BC's are implemented.

6.3.1 Mesh Statistics

Quality Index	Value
No of elements	2'129'199
Max Non-Ortho	69.12
Avg Non-Ortho	7.30
Max Skewness	4.241 (4)
Max Aspect Ratio	32.1

Table 19: Grid feature, honeycomb expansion chamber

Patch	Avg y^+
Volute	51
Rotor	10.5
Discharge	0.98

Table 20: Average y^+ values, honeycomb expansion chamber domain

For what concern the grid, based on the previous results, boundary layer cells are extruded only on the discharge section.

6.3.2 Residuals

\mathbf{p}	\mathbf{U}_x	\mathbf{U}_y	\mathbf{U}_z	\mathbf{h}	κ	ω
$9.4e-6$	$4.9e-7$	$6.9e-7$	$8.7e-7$	$2.1e-5$	$3.9e-7$	$9.1e-8$

Table 21: Initial residuals, honeycomb expansion chamber adiabatic

6.3.3 Numerical Results

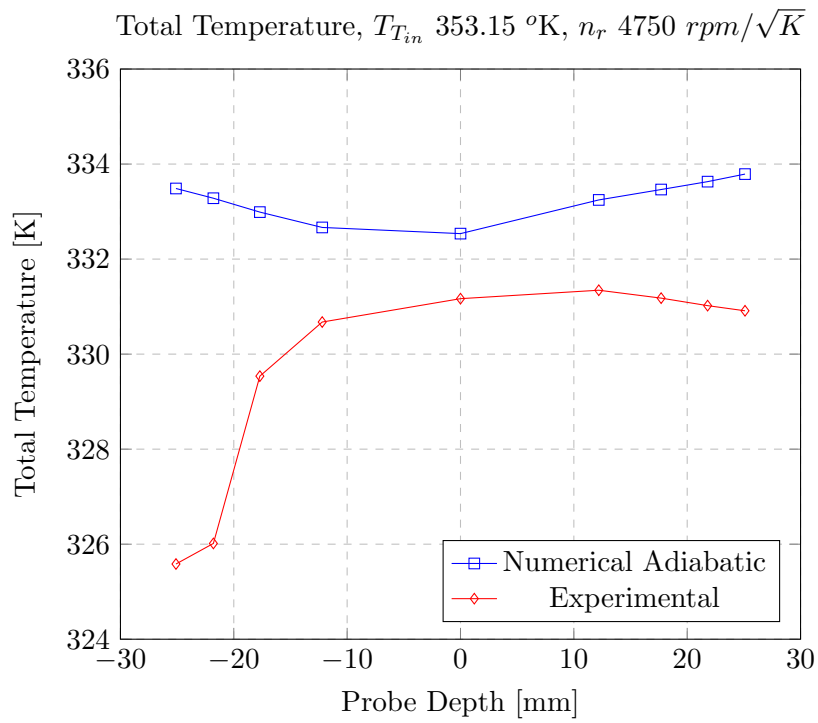


Figure 48: Total temperature, honeycomb expansion chamber, adiabatic

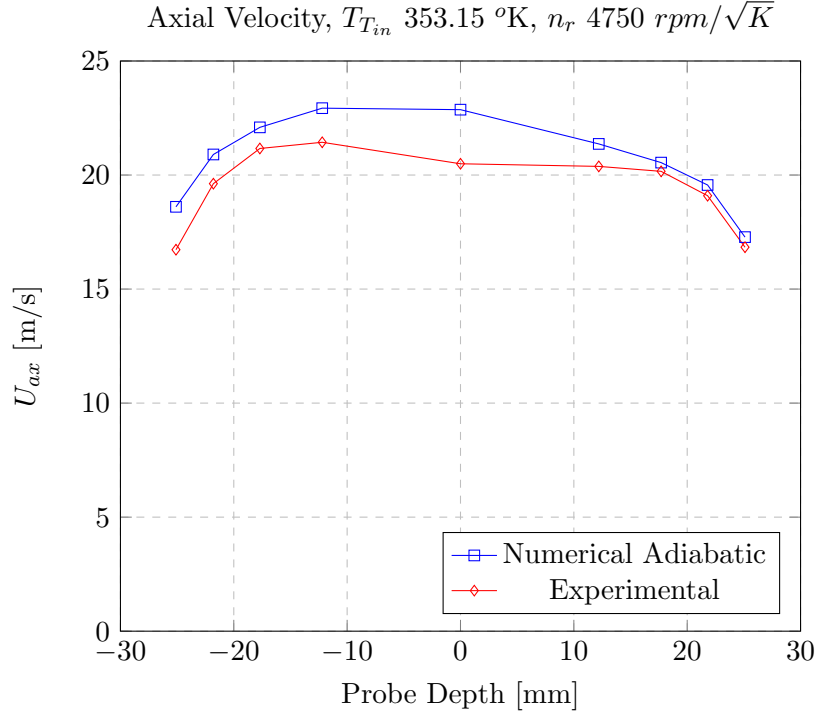


Figure 49: Axial velocity, honeycomb expansion chamber, adiabatic

Note that, with respect to the previous campaign, probing coordinates are changed so to describe circular crowns with constant area, with the aim of obtaining more accurate mass-flow averaged values. The first sampling is taken at 25.1 mm from the duct axis so at less than 2 mm from the wall. As a matter of facts, in this campaign, the effect of the measurement station on heat exchange is clearly visible. In 48, however, we could see that at the central point the difference in total temperature is more or less 1.5 K, a similar result with respect to the standard duct case. 49 states that the velocity field is computed quite accurately, especially if we take into account also that the experimental results are influenced by the probe effect on the flow.

Since main aim of the honeycomb is to eliminate the swirling component from the flow, we have to check whether this hypothesis is matched.

Fig.50, represent a 3D reconstruction of tangential velocity component. Picking fig.28 as comparison, we could actually see that swirl is almost eliminated.

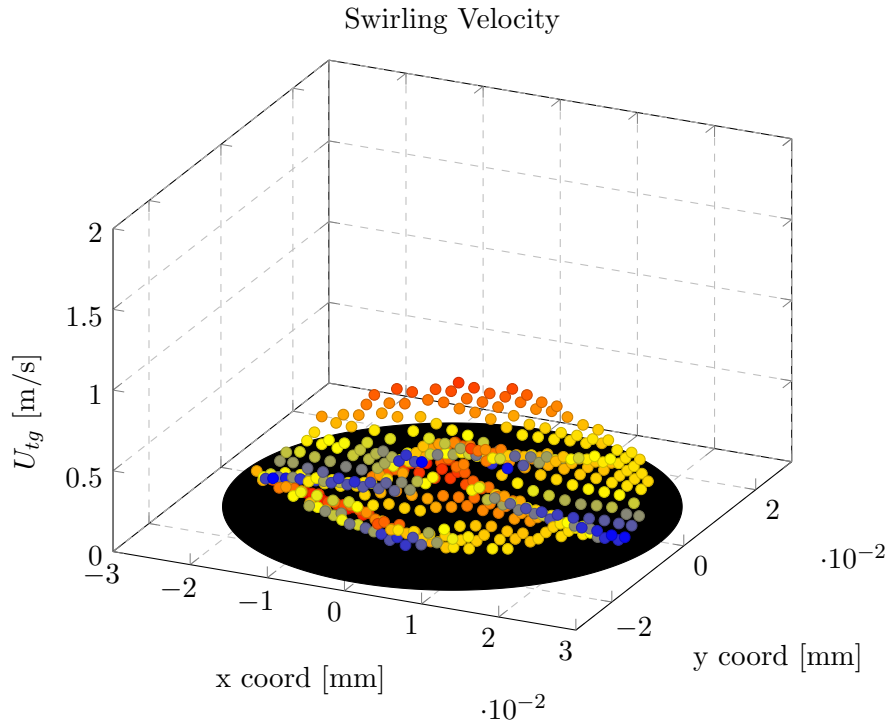


Figure 50: Tangential velocity, honeycomb expansion chamber, adiabatic

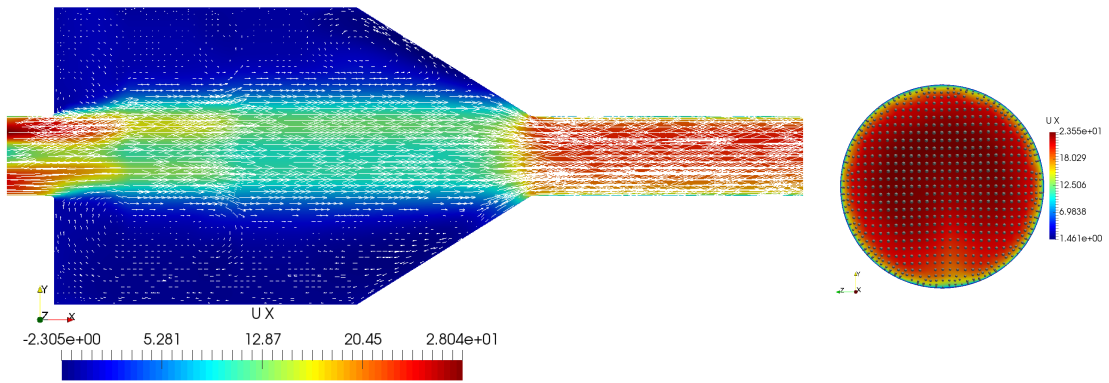


Figure 51: Axial velocity contours and vectors, cross section and measurement plane, honeycomb expansion chamber, adiabatic

6.4 Results: first operating point, addition of heat losses

As done for the standard layout, in order to improve the matching between the results, some kind of heat loss must be taken into account.

For all the cases of this experimental campaign, we have at our disposal the wall temperature at the measurement station. We will impose this latter value as fixed and uniform along the discharge section wall.

Since the best results were obtained with the uneven wall heat flux distribution, also here we will try to implement such kind of condition in the same exact way.

6.4.1 Residuals

Heat Loss	\mathbf{p}	\mathbf{U}_x	\mathbf{U}_y	\mathbf{U}_z	\mathbf{h}	κ	ω
Fix T wall	$1.2e-5$	$5.3e-7$	$8.1e-7$	$9.3e-7$	$2.6e-5$	$3.5e-7$	$8.5e-8$
Wall heat flux	$1.2e-5$	$5.1e-7$	$8.0e-7$	$9.3e-7$	$2.4e-5$	$3.2e-7$	$7.8e-8$

Table 22: Initial residuals values, honeycomb expansion chamber, first operating point

6.4.2 Numerical Results

Fig. 52 shows the trend of total temperature once the two kinds of heat exchange are activated. The fixed temperature case outputs values slightly higher than the experimental one, the main reason should be in the fact that all the upstream "hot" region it is still treated as adiabatic. Whereas once the heat flux is distributed across the whole domain, the curve falls below the experimental one but maintaining almost the same behavior. As stated before, the value of heat power to be dissipated, comes from an empirical reasoning far from the truth of facts.

We have on purpose recalled also the adiabatic result, as a matter of facts it is possible to observe that heat losses do not influence the axial velocity field. All the three curves are one over the other and they pretty much confirm the experimental result.

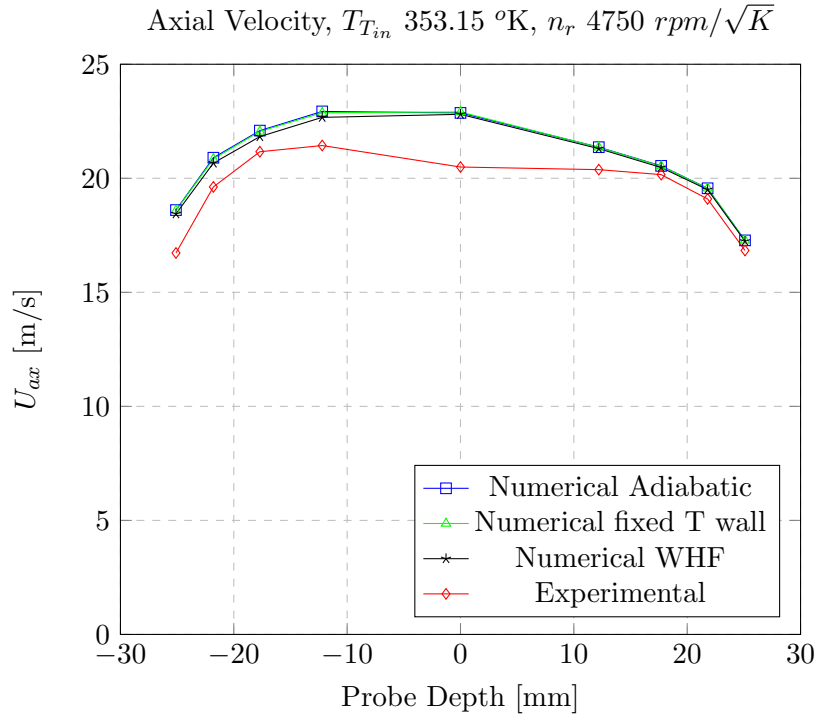
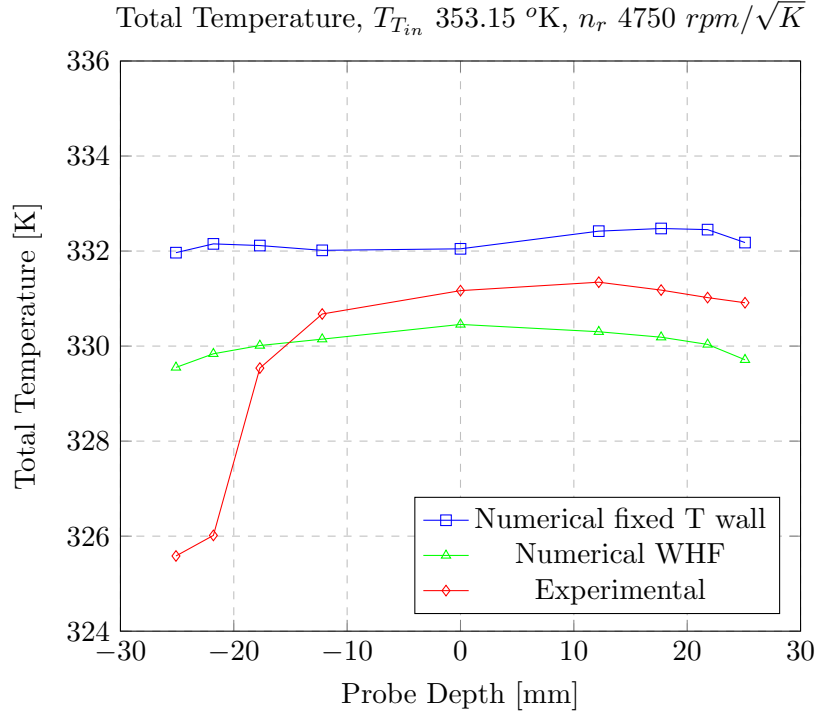


Figure 52: Total temperature and axial velocity, honeycomb expansion chamber, heat losses addition

6.5 Results: second operating point

In order to confirm the conclusion made after looking at the results of the previous chapter, we have carried out analogue simulation for another operating point:

n_{red}	$T_{T_{in}}$	\dot{m}
$5500 \text{ rpm}/\sqrt{K}$	353.15 K	0.05 kg/s

6.5.1 Residuals

Heat loss	p	U_x	U_y	U_z	h	κ	ω
Adiabatic	$2.0e-5$	$4.7e-6$	$1.2e-7$	$1.3e-7$	$2.1e-5$	$1.6e-6$	$9.9e-8$
Fix T wall	$1.3e-5$	$1.0e-6$	$3.6e-7$	$4.9e-7$	$5.6e-6$	$3.5e-7$	$8.9e-8$
Wall heat flux	$1.5e-5$	$3.9e-6$	$2.5e-7$	$3.7e-7$	$3.0e-5$	$1.7e-7$	$8.0e-8$

Table 23: Initial residuals values, honeycomb expansion chamber, second operating point

6.5.2 Numerical Results

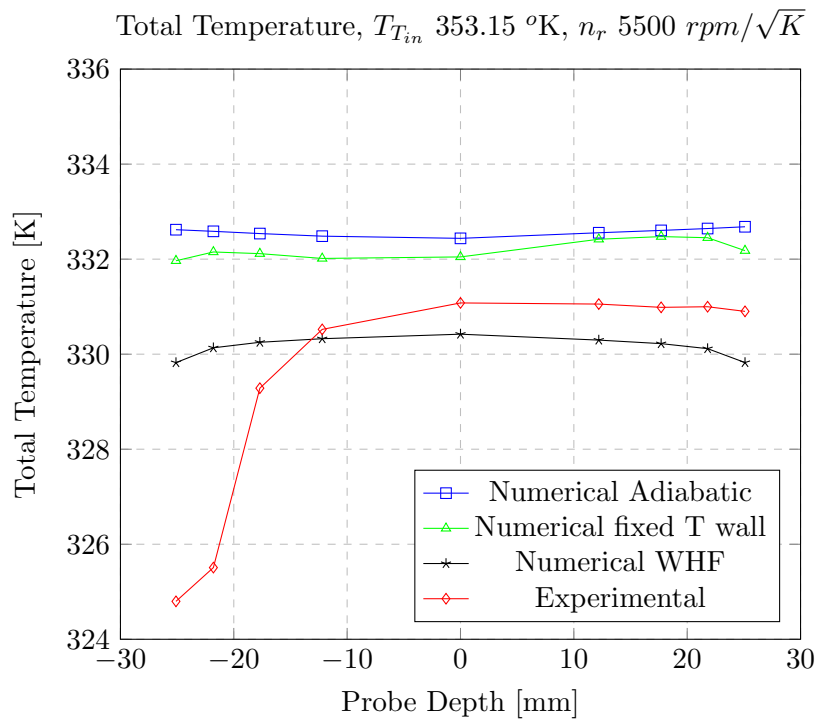


Figure 53: Total temperature, honeycomb expansion chamber, second operating point

Looking at fig.53 and 54 we could assert that, all conclusions made for the first operating point are still valid, thus the model reacts well to the variation of boundary conditions.

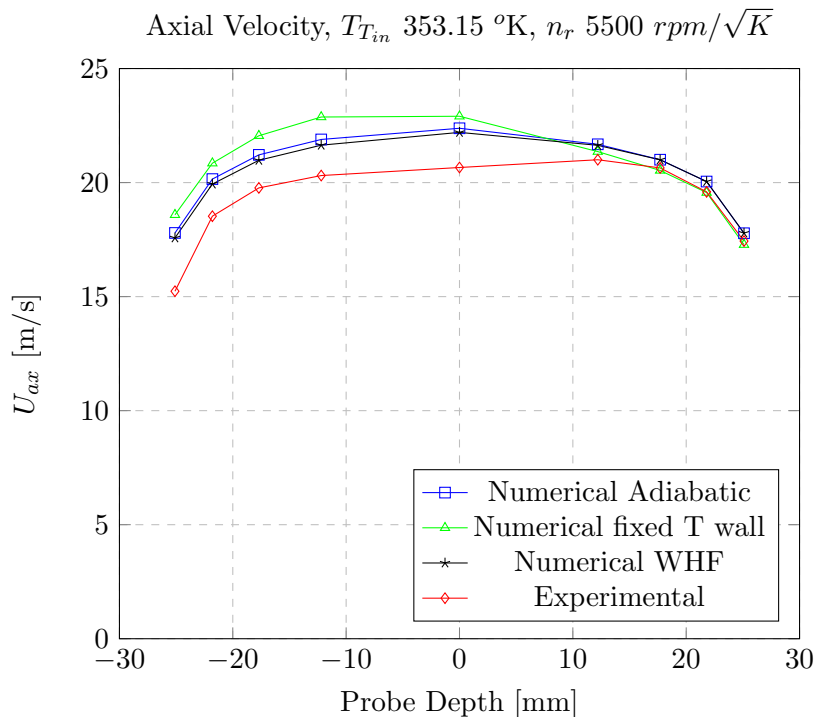


Figure 54: Axial velocity, honeycomb expansion chamber, second operating point

6.6 Grid sensitivity analysis

As already done for the standard duct, we have to make sure that output of the numerical simulation does not depend on the number of elements used to discretize the physical domain. The refinement procedure consist in halving base cell edge's length while maintaining local refinements equal with respect to the standard grid, same that has been done also for the standard duct case. In these analysis it is important to exclude the effect of the boundary layer solution method on the results. As we have previously explained, our turbulence model solve the modeled balances at the wall if the a-dimensional distance y^*+ is sufficiently small. To maintain this parameter unchanged, the dimension of the BL cells has been kept equal no matter the refinement level of the global grid.

We could see both in fig.55 that results changes consistently with the first halve of the cell size, whereas additional refinement does not lead to any further modification. Thus for all the subsequent calculations, $\Delta x/2$ grid will be employed.

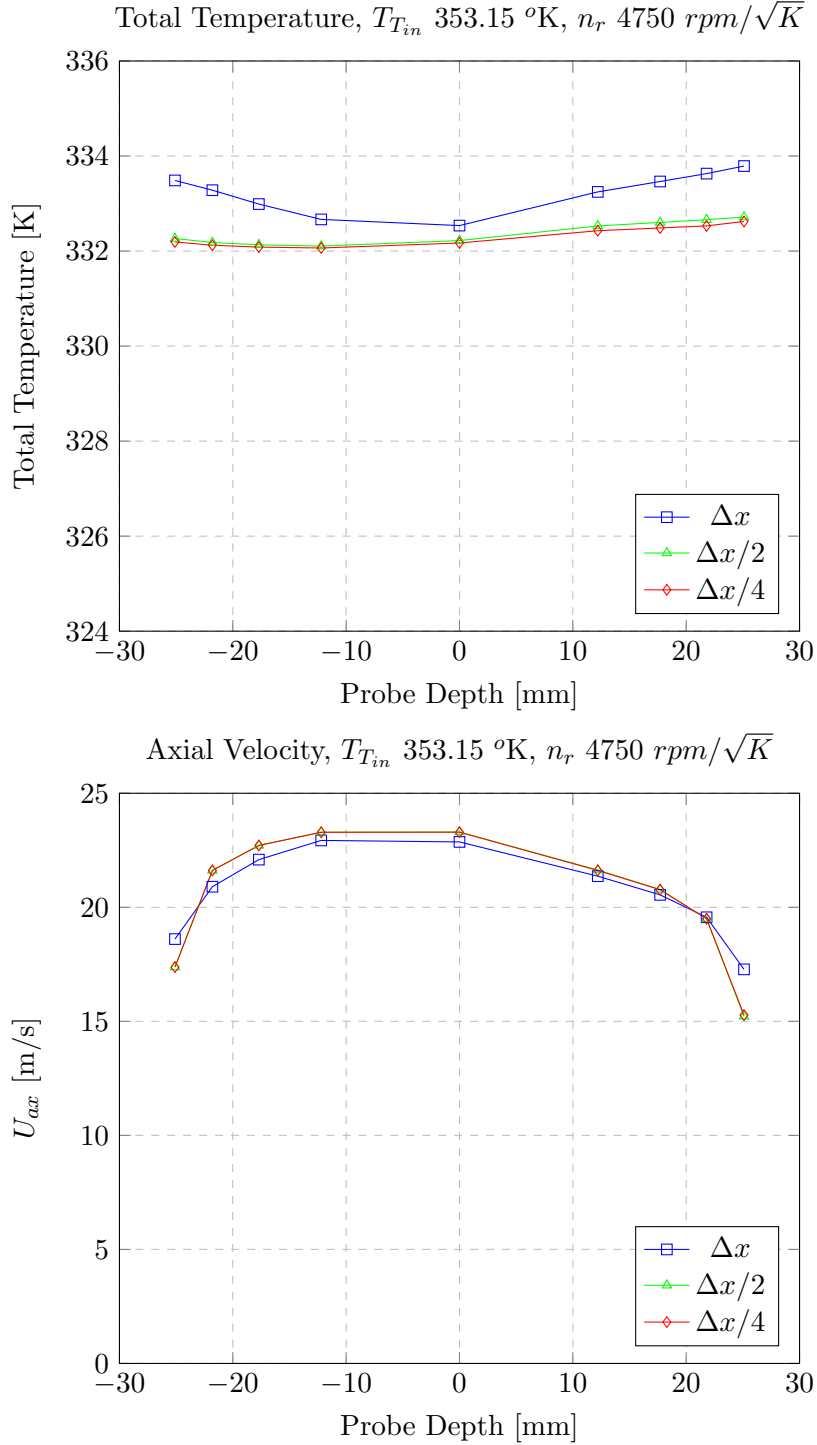


Figure 55: Total temperature and axial velocity, honeycomb expansion chamber, grid independence

6.7 Decoupling performances

As described in the pre-processing chapter, we will now need to check whether the device permits to accurately evaluate the efficiency without the need of a complex measuring campaign.

To do that, we must now proceed to the computation of γ_m , uniformity index, and to the random sampling efficiency evaluation procedure.

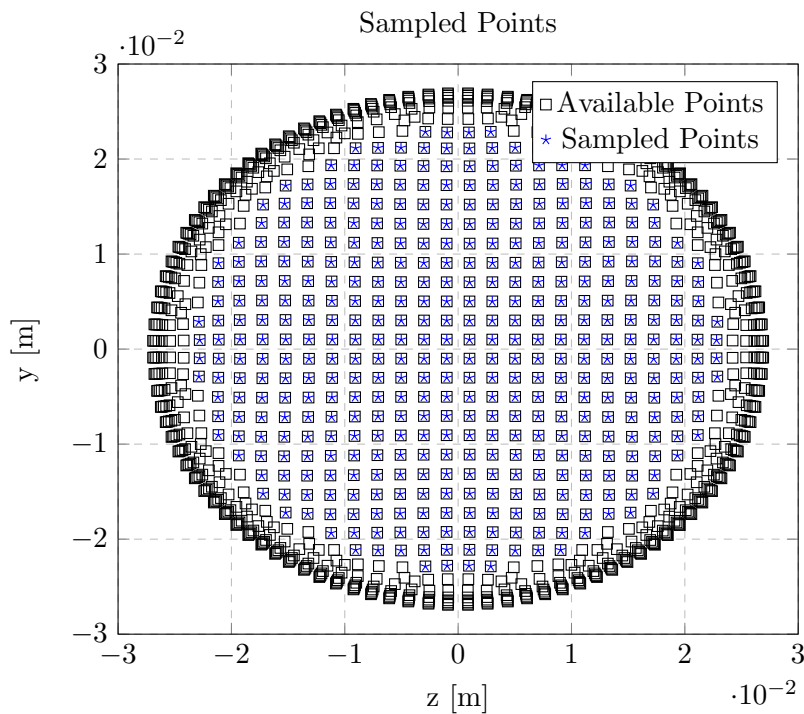


Figure 56: Sampled Points

It is important to exclude the effect of walls on these calculations. In fig. 56, the black squares represent each cell's center available on our measurement plane, whereas the blue stars show which are the ones actually used. Moreover since we have seen that heat losses does not influence the bulk flow field, all the subsequent results refers to adiabatic walls condition.

The procedure used to mass-flow average the total temperature, corresponds to the one applied to experimental campaigns but, in this case, the number of samples is way higher, thus leading to a more accurate result.

Moreover in table 24 we report performances in terms of uniformity index γ_m , based on the meridional velocity field, along with the total pressure loss caused by the flow conditioner.

6.7.1 First operating point

The following results are referred to a reduced rotational speed of $4750 \text{ rpm}\sqrt{K}$ and total inlet temperature of $80 \text{ }^\circ\text{C}$.

Grid	γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
Standard	0.9808	1310	0.1706
Refined	0.9776	1068	0.2983

Table 24: Performances indexes, honeycomb expansion chamber, $n_{red} 4750 \text{ rpm}\sqrt{K}$

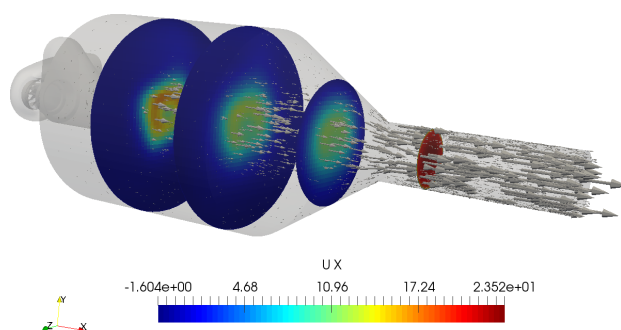


Figure 57: Graphical post-processing, honeycomb expansion chamber, axial velocity contours

In fig. 58 we could observe the results: black line corresponds to mass-flow average, whereas the other marks refer to 100 random sampling campaigns. It is clear to see that when we shift from a single sampled (black) point to three samples (green) the resulting value gets closer ($\pm 0.5\%$) to the "real" weighted average.

We could say that the device's performance, in terms of the ratio $\gamma_m/\Delta p_T$, is already a good benchmark. As a matter of fact, during the experimental campaign, the efficiency value computed through the local probing procedure was only 1.9% higher with respect to the one coming from the standard thermocouples readings. Moreover, when passing to the finer grid, we obtain lower losses. An explanation could rely in the fact that decreasing element size reduces the numerical viscosity associated to upwind schemes. Actually, for the same reason, also the efficiency value sees an increase of almost 3%. As a consequence, the random sampling pattern is concentrated in a narrower interval. However, looking at the velocity distribution (fig.57), we could see that this geometry leads to very high flow recirculation areas. The key point of such a device is the ability of reducing the flow velocity as much as possible before entering the honeycomb matrices. Thus at this point we could think of a diffuser able to reduce as well the velocity but without flow detachment related losses.

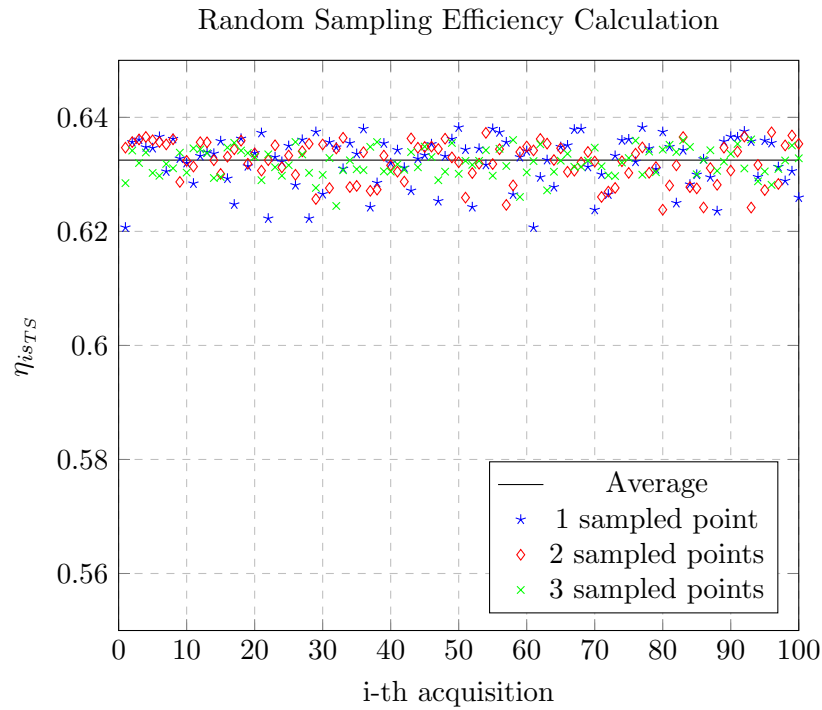
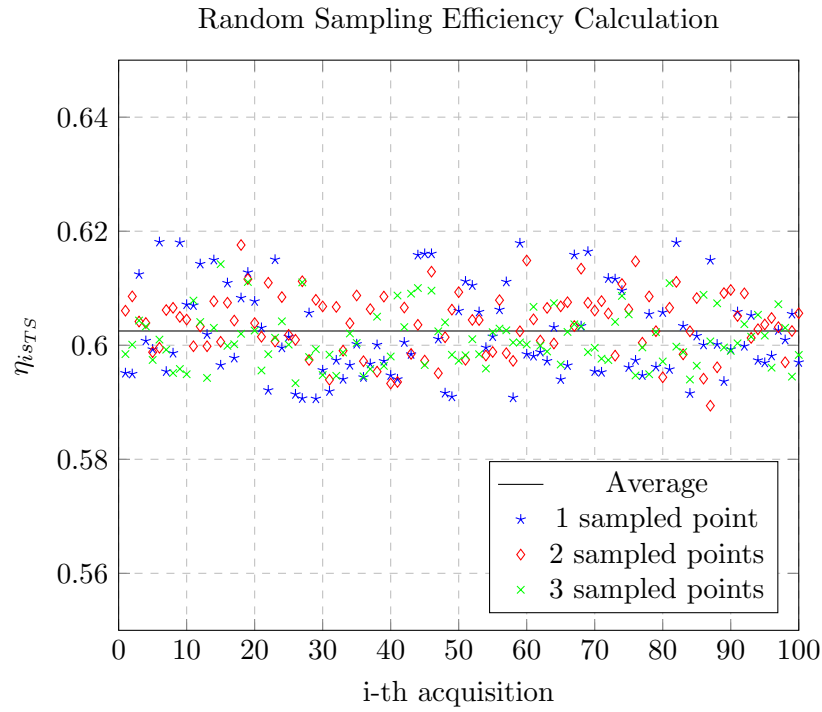


Figure 58: Efficiency, expansion chamber standard (top) and refined (bottom) grids

6.7.2 Second operating point

To prove the goodness of this device, we must now simulate another turbine operating point in order to exclude the influence of a particular set of conditions on results. Since we have already available data regarding $5500 \text{ rpm}\sqrt{K}$ iso-velocity, we may now compute performances indexes.

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9843	1679	0.08

Table 25: Performances indexes, honeycomb expansion chamber, $n_{red} 5500 \text{ rpm}\sqrt{K}$

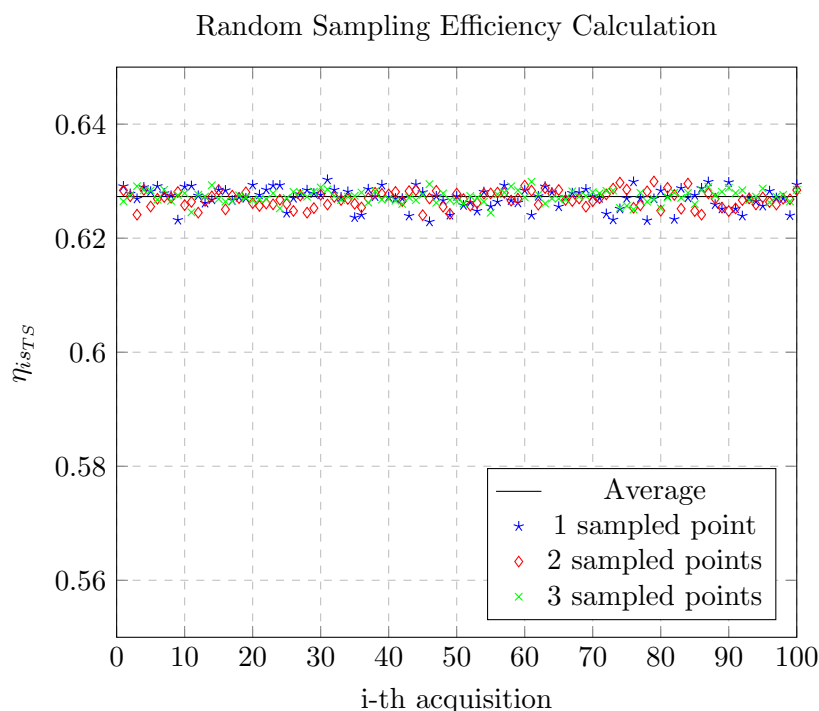


Figure 59: Efficiency, expansion chamber, $n_{red} 5500 \text{ rpm}\sqrt{K}$

Since we will pick the first operating point as reference case, only numbers associated to standard grid are here reported. With respect to the previous result, the latter shows a better uniformity both in terms of axial velocity and total temperature. On the contrary, pressure loss is higher, given by the fact that to an increased rotational speed also corresponds an higher discharged velocity.

In conclusion, this flow conditioner has been proved reliable also for such conditions.

6.8 Influence of turbulence treatment

As we have depicted in section 3.5.3, a deep analysis of honeycomb effects on turbulence has been carried out. Thus in this section we are going to establish if, adding source and sink terms to generate or destroy turbulence, has any influence or whatsoever on results. Reference case for this study is the first operating point in adiabatic conditions applied to standard grid.

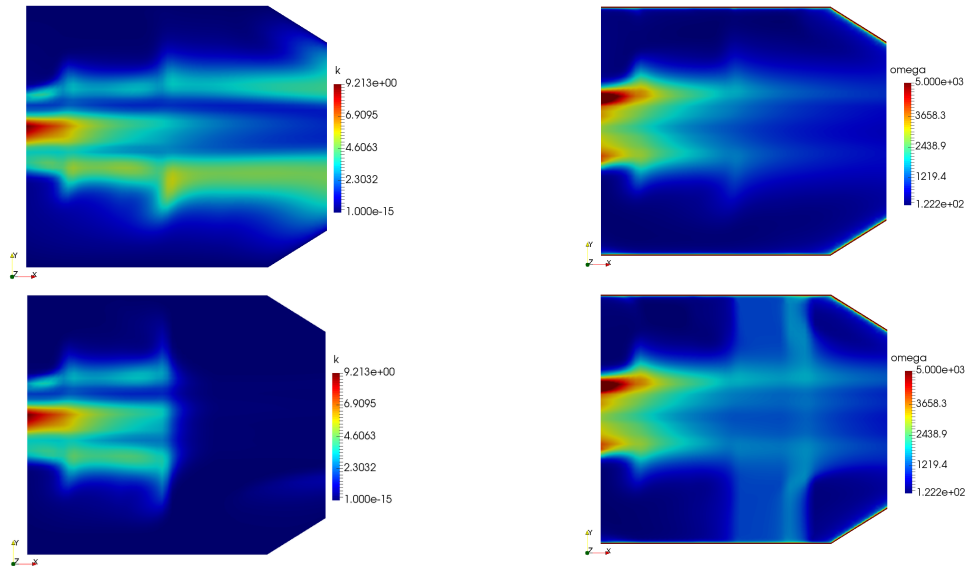


Figure 60: Turbulent kinetic energy and specific frequency contours, without (top) and with source (bottom)

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9796	1300	0.2125

Table 26: Performances indexes, honeycomb expansion chamber with turbulence sources

In the figure above, we could see that adding the source and sink terms reflects quite heavily on turbulence quantities, however in terms of macro values (table 26) the uniformity γ_m decreases of 0.12% along with a meaningless 10 Pa decrease in pressure drop. As a matter of facts, we have decided to not include this treatment in the future calculation in order to lighten up the computation.

6.9 Influence of expansion chamber diameter

As a first attempt in trying to improve the performances of this device, we could try to see what happens if we vary the plenum diameter. As a matter of facts, two new geometries were generated (fig.61). Keeping into mind that original chamber diameter is 200 mm , the first features a reduced volume chamber ($\Phi_{chamber} 150\text{ mm}$), whereas the latter foresees an increased one ($\Phi_{chamber} 250\text{ mm}$). The honeycomb matrices' positions and thicknesses were left unaltered.

The spatial discretization relies on the same features of the standard one, if then we will see some kind of improvement, we will shift towards a more refined one.

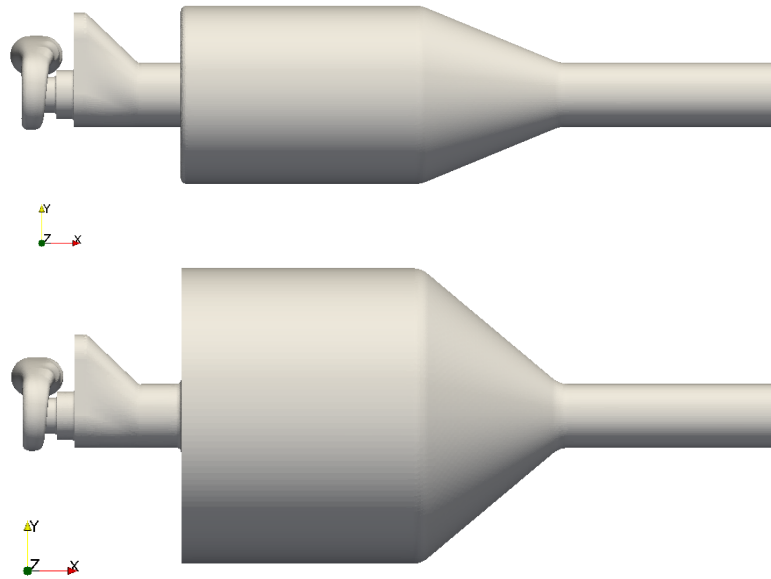


Figure 61: Reduced (top) and increased (bottom) expansion chamber domains

6.9.1 Results

Chamber	p	U_x	U_y	U_z	h	κ	ω
Small	$1.2e-5$	$4.3e-6$	$3.1e-6$	$3.3e-6$	$2.6e-5$	$2.7e-7$	$8.3e-8$
Big	$2.6e-5$	$4.9e-6$	$3.4e-6$	$3.6e-6$	$3.7e-5$	$3.2e-7$	$1.6e-7$

Table 27: Initial residuals values, reduced and increased expansion chamber volume

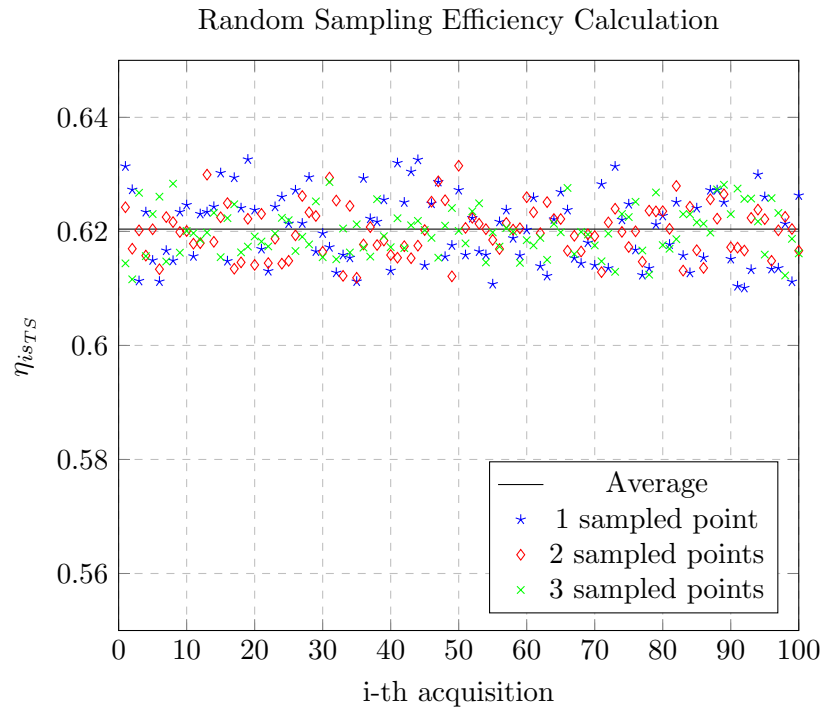
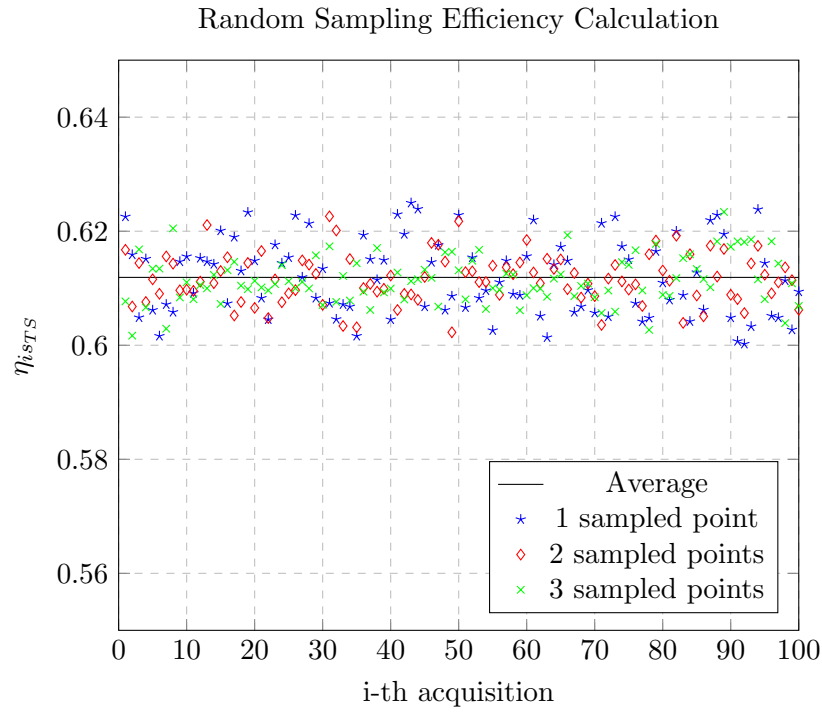


Figure 62: Efficiency, reduced(top) and increased (bottom) expansion chamber

Chamber	γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
Small	0.9791	1870	0.3069
Big	0.9824	2316	0.2542

Table 28: Performances indexes, reduced and increased expansion chamber

In fig.62 it is reported, as for the previous case, the results obtained with the random sampling efficiency calculation. In particular we could see that, for the smaller chamber, the values are more widely spread around the "true" mean value, thus indicating a worsening of mixing performances. On the other end, for the increased expansion volume, uniformity is slightly better. However, looking at the performance indexes, we have to highlight the big increase in total pressure loss for both cases.

So, to conclude, we have found out that increasing the chamber volume leads to better flow uniformity, but at the same time, increases the losses as well. With the original geometry, that stays in between the two tested scenarios, achieving the better trade off we could now fix this degree of freedom and carry on with further analysis.

6.10 Single honeycomb matrix

Another aspect worth to be considered is identified in the possibility of adopting a single honeycomb matrix instead of a couple. For the first trial, thickness and positioning of the media is maintained constant with respect to the reference case. We have only eliminated the second matrix and carried out the calculation.

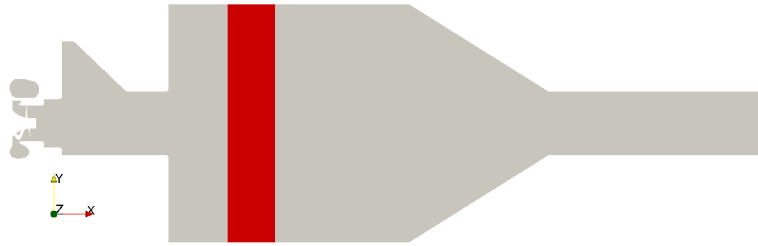


Figure 63: Single honeycomb expansion chamber domain

6.10.1 Results

Since outputs obtained with the standard grid setup showed a slight improvement from the reference 2 matrices case, we have decided to carry on also another calculation with a refined grid, generated halving base cell edge's length. In this chapter, both results are reported.

Grid	p	U_x	U_y	U_z	h	κ	ω
Standard	$4.2e-5$	$3.2e-6$	$3.2e-6$	$4.0e-6$	$1.7e-5$	$3.1e-6$	$2.4e-7$
Refined	$2.7e-5$	$6.8e-6$	$4.8e-6$	$5.4e-6$	$1.4e-5$	$9.2e-6$	$1.1e-7$

Table 29: Initial residuals values, single honeycomb expansion chamber

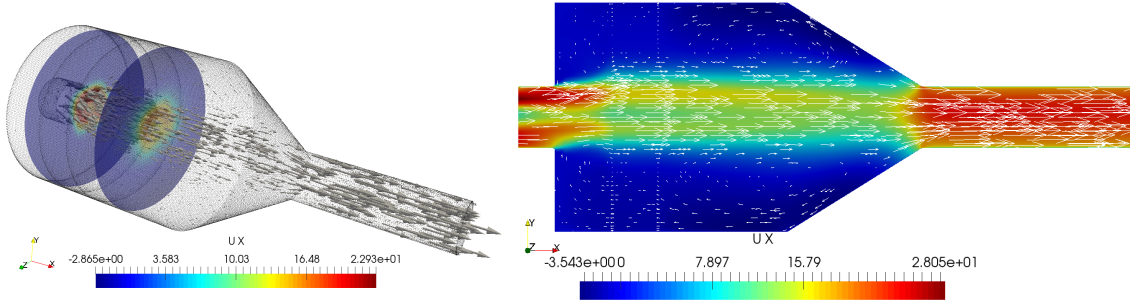


Figure 64: Graphical post processing, axial velocity contours, single HC expansion chamber

In fig.64, the contours of axial velocity are plotted in different sections. If we compare it with results coming from double honeycombs, we can not see much of a difference. In fact we have obtained less total pressure drop, while maintaining more or less the same flow uniformity. The result changed dramatically when shifting toward refined mesh. On the contrary of what we have seen for the double honeycomb case, we notice a slight increase in uniformity along with a big 500 Pa increase in pressure drop (table 30). We can draw the conclusion that 2 matrices acts as a sort of stabilizer avoiding excessive recirculation thus, at the end, reducing back-pressure.

As a matter of fact, the original layout of this flow conditioner has revealed itself as the most proficient one.

For what concern positioning of matrices inside the expansion chamber, we could say that the first one must be placed relatively close to the outflow. That is because we need to eliminate swirling component as soon as possible (see radial equilibrium issue in next chapter). For this reason we have chosen not to deepen this aspect and to leave frozen this degree of freedom.

Grid	γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
Standard	0.9793	1279	0.1577
Refined	0.9804	1987	0.2272

Table 30: Performances indexes, single HC expansion chamber

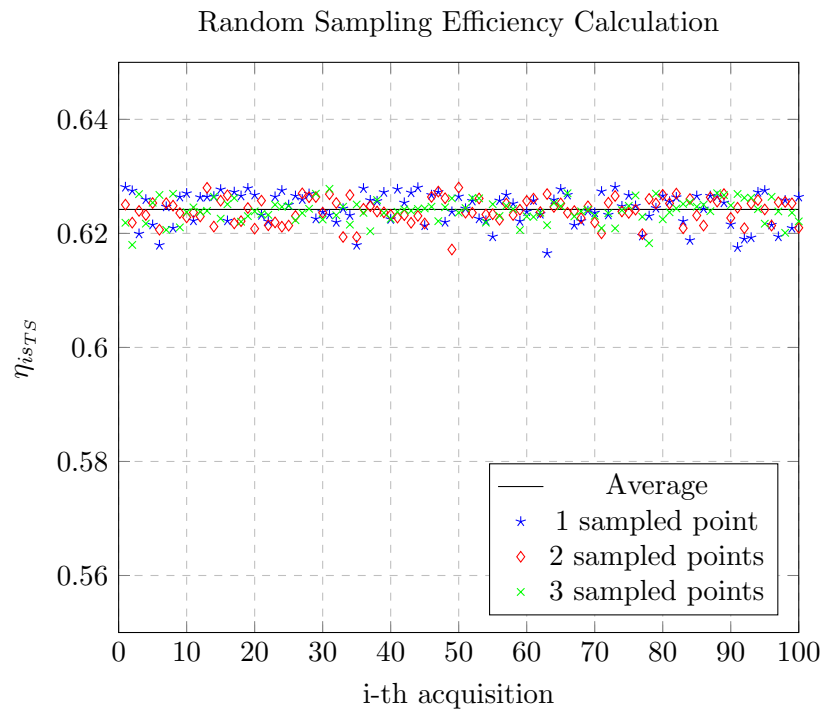
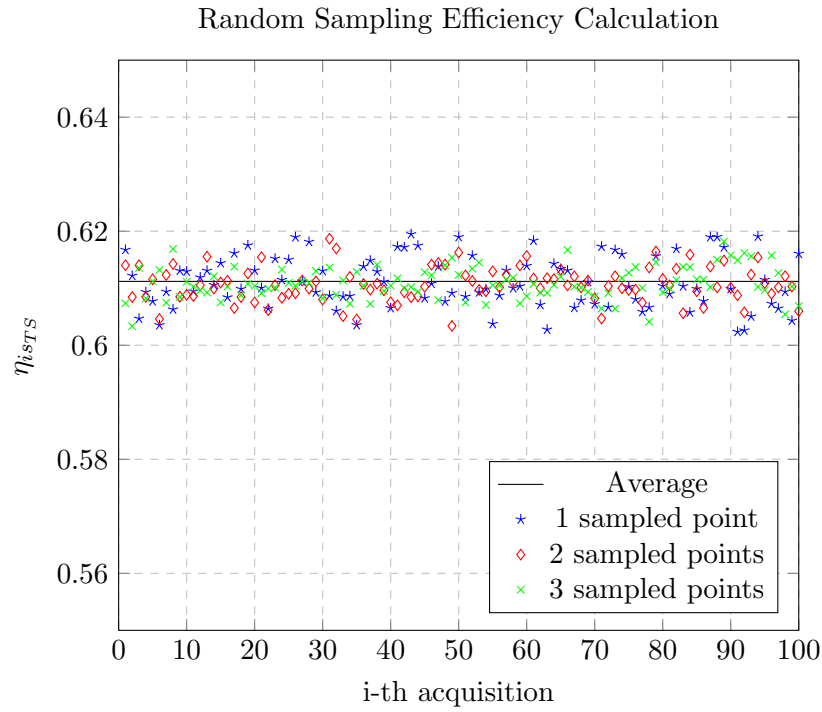


Figure 65: Efficiency, single HC, standard (top) and refined (bottom) grid

7 Diffuser decoupler

7.1 Diffusing related issues

Before start designing a new geometry, we have to take into account issues related to the diffusion of such an high swirling flow field. A flow that posses a non negligible tangential component undergoes the phenomena of the so-called radial equilibrium. This states that since a centrifugal force arises, a pressure gradient directly proportional to the radial distance from the duct axis develops (see fig. 66). This force is generated by walls that in this way contrast the radial volume shift given by the centrifuge.

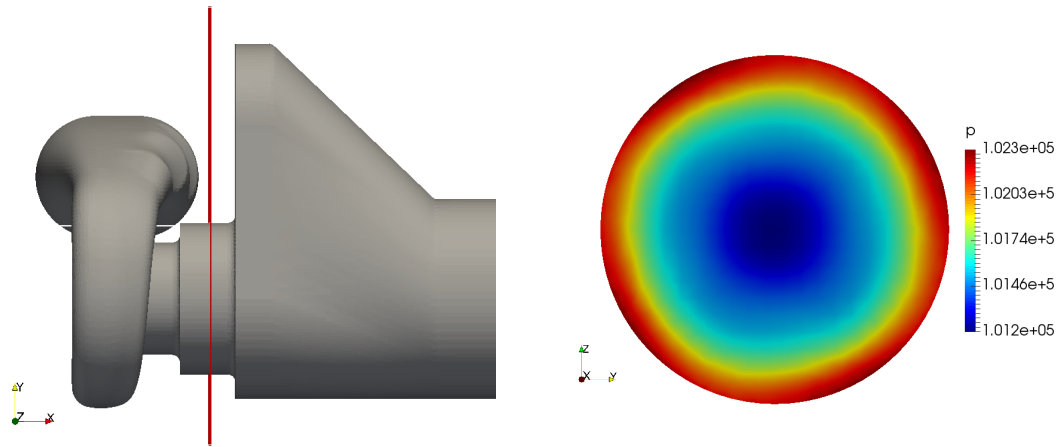


Figure 66: Radial Equilibrium

Figure 67 refers to an expansion chamber without honeycombs, after outflowing in the casing, there's nothing left to balance the centrifugal potential, thus the combination of axial and tangential momentum leads to an helical like evolution.

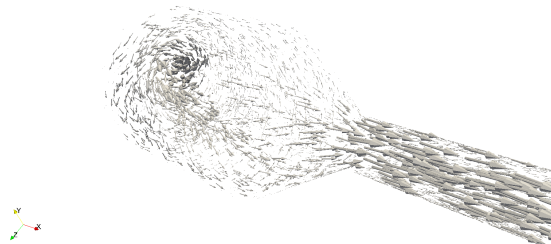


Figure 67: 3D velocity vector inside decoupler's casing

At this point we could imagine to insert a traditional linear diffuser. The only way to keep the flow attached, would be in the case of a really small aperture ($2-3^\circ$) that will result in a ridiculously long device to obtain an acceptable pressure recovery. On the other end, in case of a canonical diffusion angle of $6-8^\circ$, we expect the onset of similar detachment and recirculation problem.

7.2 Geometry and meshing

The first constraint we have to satisfy is the total length of the system, as a matter of facts, the measurement section must stay within the same axial coordinate of the expansion chamber layout.

In fig. (68) we could see the technical drawing of the new device. In particular we have a diffusing length of 2 and a half diameters along with an aperture of 7° , then 2 diameters length at constant section (where the honeycomb will be placed) and, finally, the convergent is the exact copy of the divergent. The red section represents where the honeycomb matrix will be inserted.

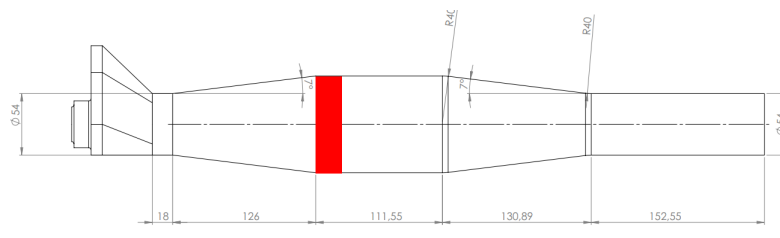


Figure 68: Technical drawing of the diffuser geometry

For what concern grid generation, the same basic parameters are maintained. As for the previous cases we're going to refine the discretization only if we will obtain good outputs.

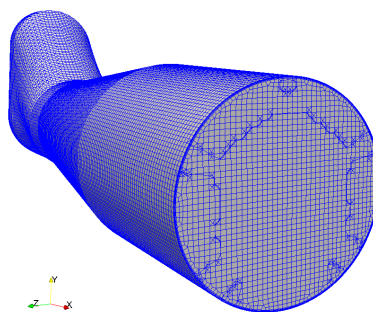


Figure 69: Mesh of the diffuser discharge

7.3 Results

As a starting point, we are going to evaluate what happens without inserting any kind of flow conditioner inside the diffuser casing. Afterwards, we could insert a small honeycomb matrix and evaluate the performances as before, in terms of γ_m and Δp_T .

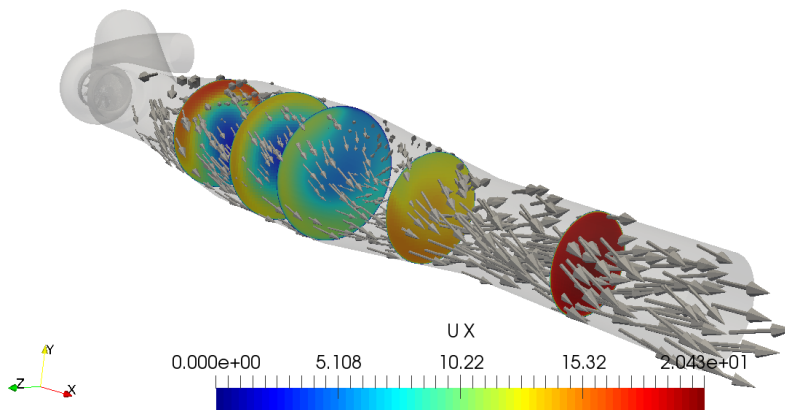


Figure 70: Graphical post-processing, diffuser decoupler, axial velocity contours

As predicted from the theory basis, the diffusing action is not efficient at all. In particular, looking at the cross sections in fig.70, we see that the biggest portion of the flow-rate, follows the ideal helical path driven by the centrifugal momentum hence leading to big recirculation zones. At this moment we're going to evaluate if the insertion of a honeycomb will improve performances. Results are depicted in fig.71. We could see that the honeycomb acts as a sort of stabilizer. As a matter of facts, it eliminates the swirling component but, flow-rate distribution is again influenced by what happens upstream. Looking at performances indexes (table 32), total pressure loss increases with respect to the reference case, along with a uniformity " γ_m " decrease. Moreover in fig.72, the result of random sampling efficiency calculation, produces a very spread pattern. Indeed such a geometry won't lead to any benefits with respect to the original layout.

\mathbf{p}	\mathbf{Ux}	\mathbf{Uy}	\mathbf{Uz}	\mathbf{h}	κ	ω
$6.8e-5$	$2.1e-6$	$4.1e-6$	$5.4e-6$	$1.8e-6$	$1.2e-6$	$1.5e-7$

Table 31: Initial residuals values, honeycomb diffuser

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9760	1823	0.2460

Table 32: Performances indexes, honeycomb diffuser

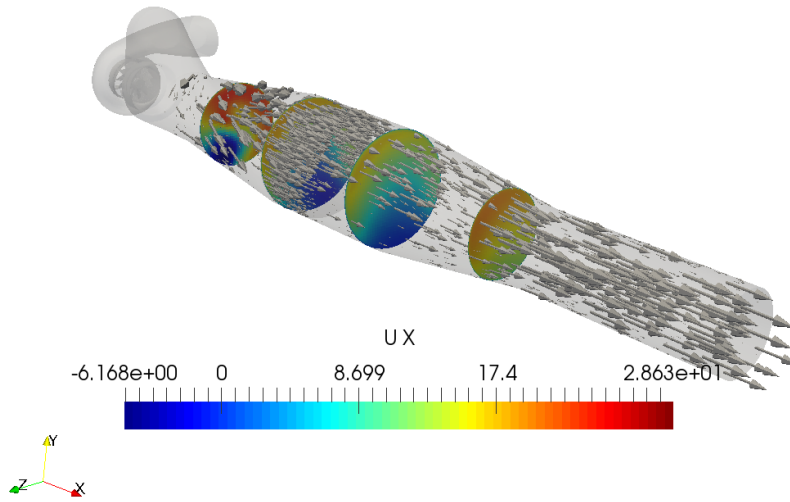


Figure 71: Graphical post-processing, honeycomb diffuser, axial velocity contours

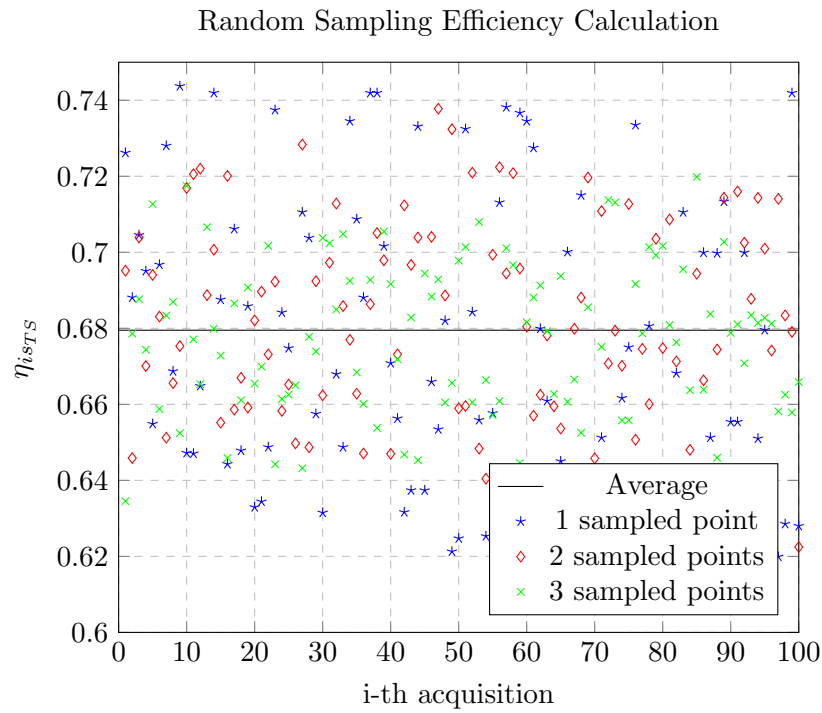


Figure 72: Efficiency, honeycomb diffuser

8 Asymmetrical diffuser decoupler

A possible solution to an efficient diffusing action could be to exploit the centrifugal potential to keep the flow attached to walls. As a matter of facts, it is a procedure widely used in conical diffuser for aero-engines ([14]). That means producing a non-axis-symmetric device.

8.1 Geometry and meshing

If we look at our turbine from the outlet section, it rotates in counter-clockwise direction. Referring to the diffuser cross section as a watch dial, we could say that the volume tends to shift towards h. 11:00. Thus we could try to diffuse only in the positive y direction we would have the flow momentum helping the process. Honeycomb matrix location is shown in red (fig.73).

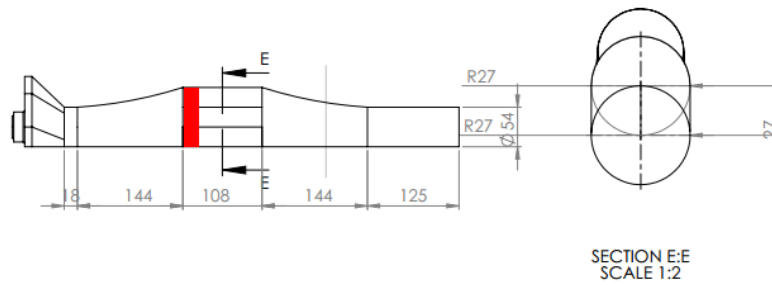


Figure 73: Technical drawing, asymmetrical diffuser

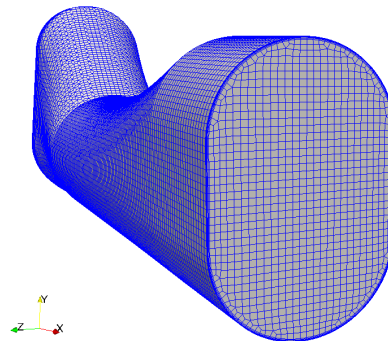


Figure 74: Mesh, asymmetrical diffuser

8.2 Results

In fig.75, we could see that we do not have any recirculation or detachment issue but, at the same time, the axial velocity distribution is not completely uniform when diffusing.

\mathbf{p}	\mathbf{U}_x	\mathbf{U}_y	\mathbf{U}_z	\mathbf{h}	κ	ω
$7.7e-6$	$3.8e-6$	$2.5e-6$	$2.1e-6$	$3.8e-6$	$7.5e-7$	$1.3e-7$

Table 33: Initial residuals values, honeycomb diffuser

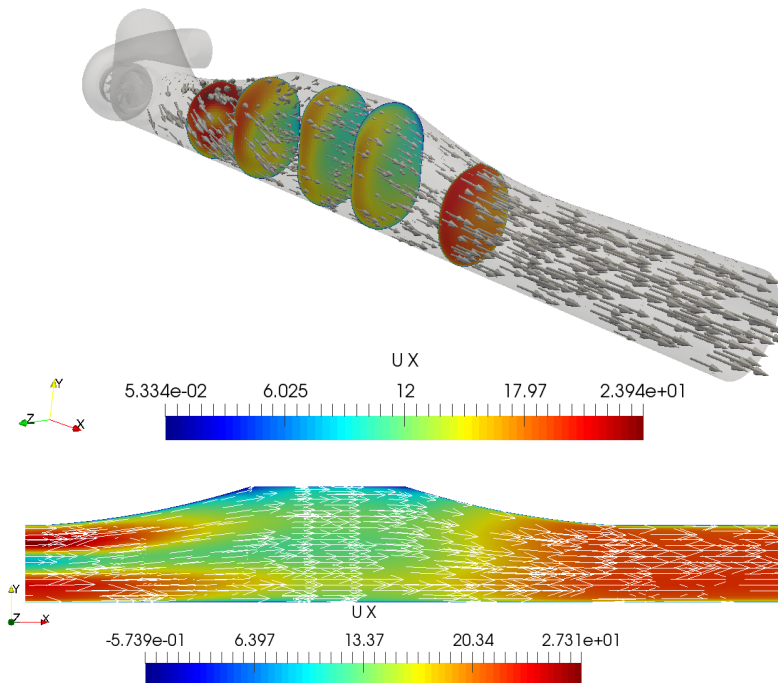


Figure 75: Graphical post-processing, asymmetrical diffuser, axial velocity contours

In terms of performances, the uniformity at the measurement section has improved (from 0.9808 to 0.9842), but the pressure loss has increased as well (1310 to 1700). This is mainly given by the higher honeycomb passing velocity.

The better homogeneity is also confirmed by the random sampling efficiency computation in fig.77. In particular, already with 2 sampling point arithmetically averaged it is possible to obtain a quite reliable result. This will give us enough motivation to study the grid influence on results. Another aspect that we have not taken into account yet, is the fact that these figures are strongly dependent on the operating point of the turbine itself. For example, maybe at lower flow-rates we do not have enough momentum to exploit thus leading again to flow detachment. However, the high value of γ_m make this solution worth to have a trial with a more refined grid, in order to see the effect of this aspect on the outputs. As for the other cases, the first refinement step consist of halving the base cell edge's length while maintaining same dimensions for local refinements regions (fig.76).

Actually, along with this latter procedure we found a worsening of performances, both in uniformity and pressure losses. Thus, in conclusion, there is not enough evidence that motivates an actual physical experimentation of this layout.

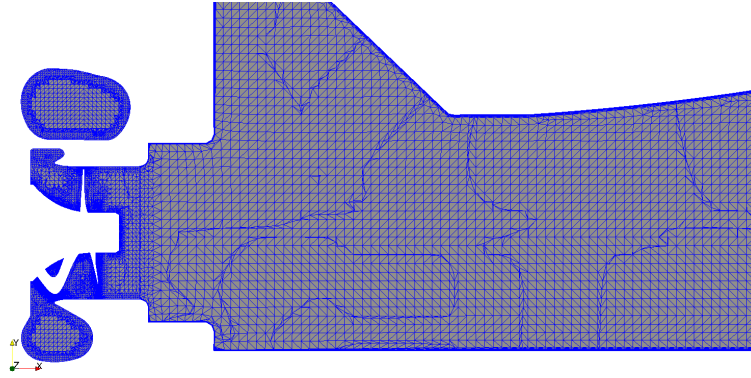


Figure 76: Cross section detail of the refined grid

Grid	γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
Standard	0.9842	1700	0.2490
Refined	0.9760	1835	0.3629

Table 34: Performances indexes, asymmetrical honeycomb diffuser

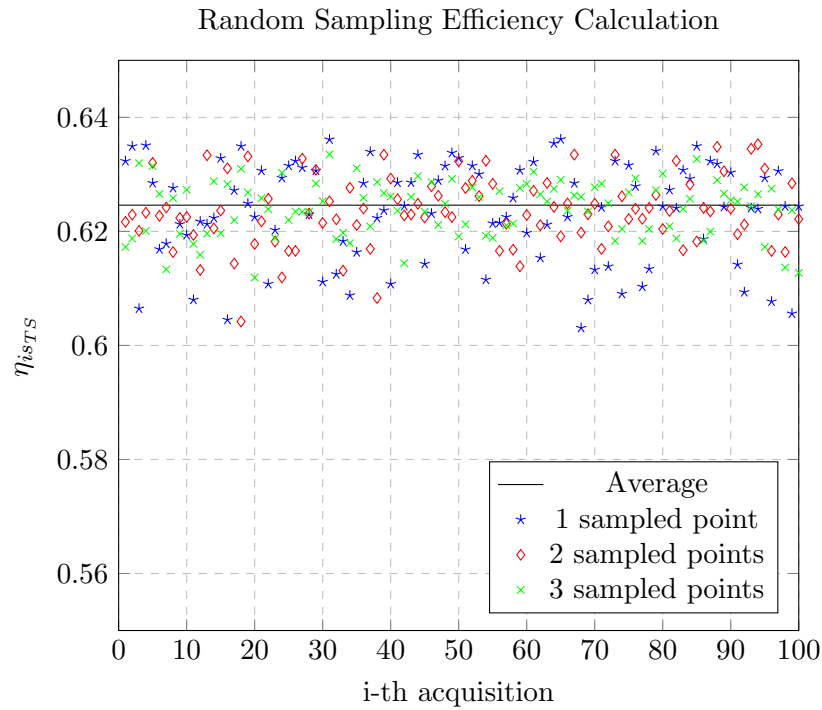
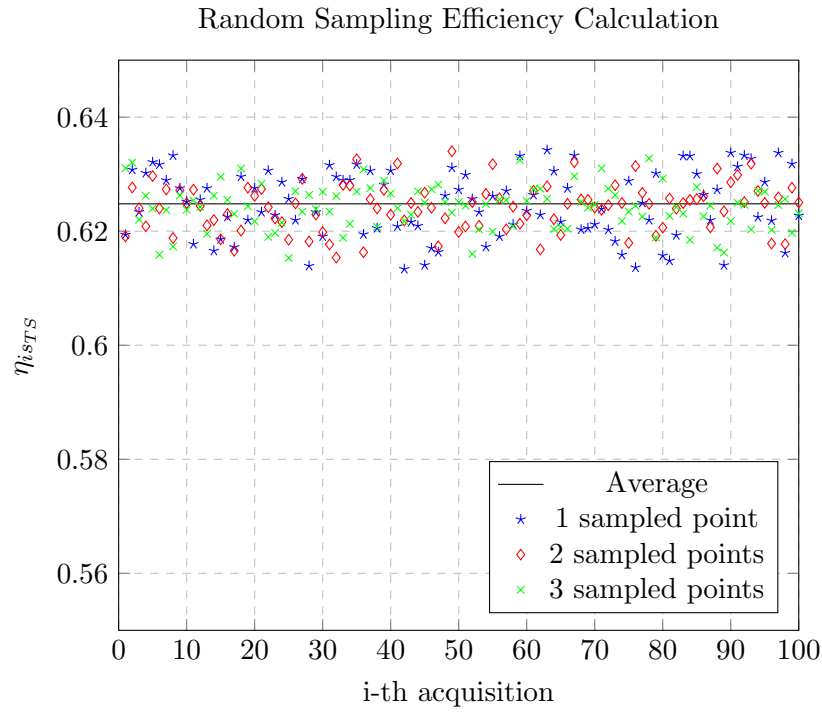


Figure 77: Efficiency, asymmetric diffuser, standard grid (top) and refined (bottom)

9 Honeycomb mixer decoupler

At this point after having assessed that an efficient pressure recovery is almost impossible we could try to investigate the possibility of homogenizing the flow inside a constant diameter duct. The honeycomb comes in our help since, as far as we know, it is the most efficient de-swirler.

Moreover looking at fig.78, we see that somehow we have to act on the uniformity of the axial velocity field.

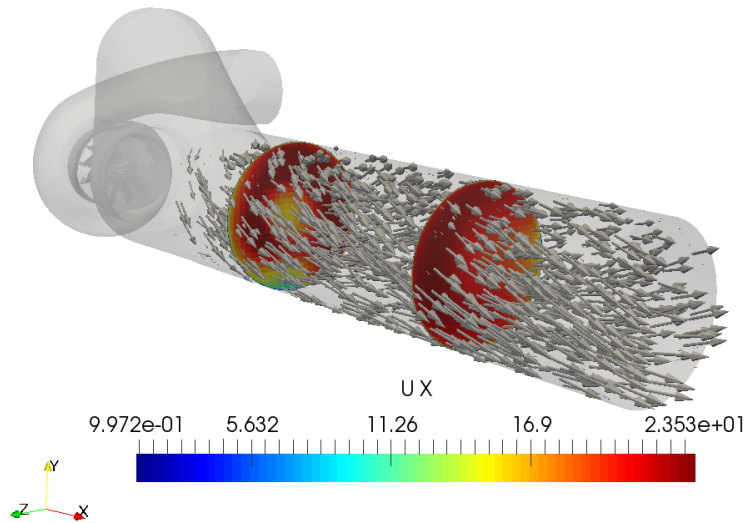


Figure 78: Graphical post-processing, standard duct, axial velocity contours

9.1 Geometry and meshing

The basic idea was to introduce some sharp fins (fig.79) just after the waste-gate turbine discharge. This could provide some additional mixing, especially in the radial direction. Then after having left some space to smooth out the flow profile, a small (10mm) honeycomb is inserted (see red zone in fig.80).

It is straightforward that losses coming from the porous media will be higher (higher passing velocity) but, adopting a constant diameter duct also means avoiding any possible detachment issue.x

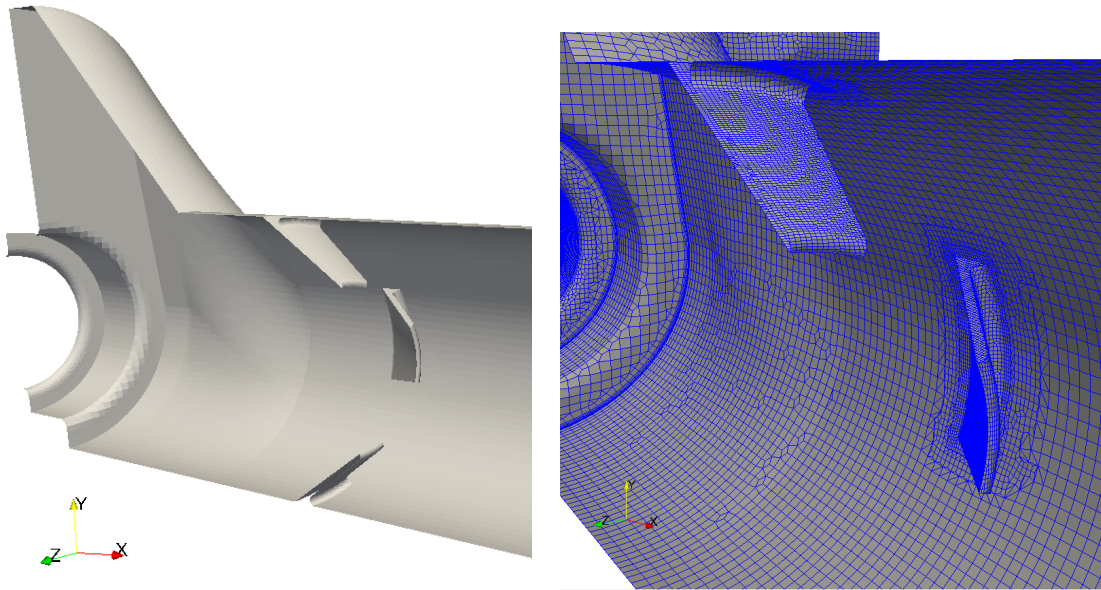


Figure 79: Detail of the mixing fins and their meshing

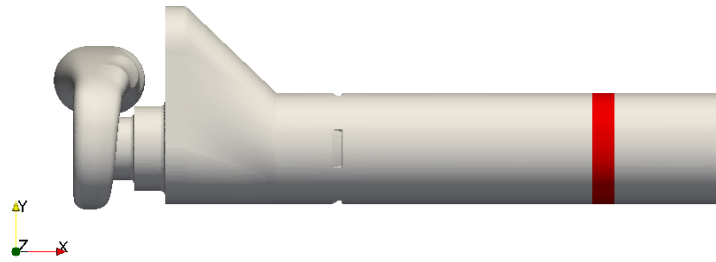


Figure 80: 3D drawing, honeycomb mixer decoupler

9.2 Results

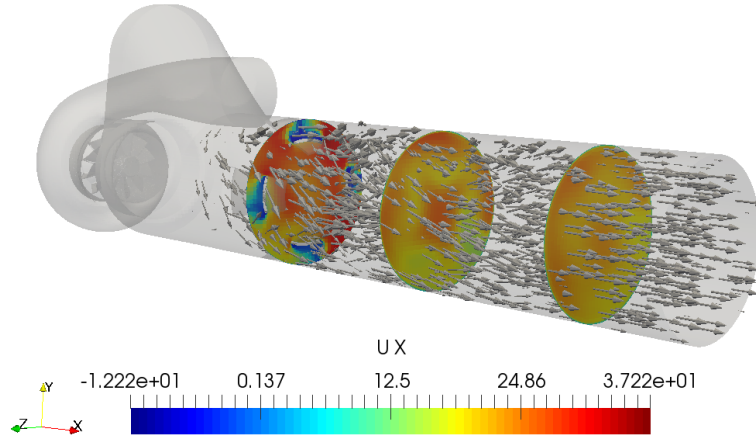


Figure 81: Graphical post-processing, mixer honeycomb decoupler, axial velocity contours

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9743	1650	0.0705

Table 35: Performances indexes, honeycomb mixer

From the graphical post processing in fig.81, we could say that our intents are matched. However if we look at the numerical outputs both in term of performance indexes and efficiency calculation, we have obtain worse result with respect to the reference case. Thus even this solution is not suitable for our purposes.

\mathbf{p}	$\mathbf{U_x}$	$\mathbf{U_y}$	$\mathbf{U_z}$	\mathbf{h}	κ	ω
$1.7e-5$	$2.7e-6$	$3.6e-6$	$2.7e-6$	$2.1e-6$	$9.7e-7$	$7.8e-8$

Table 36: Initial residuals values, honeycomb mixer

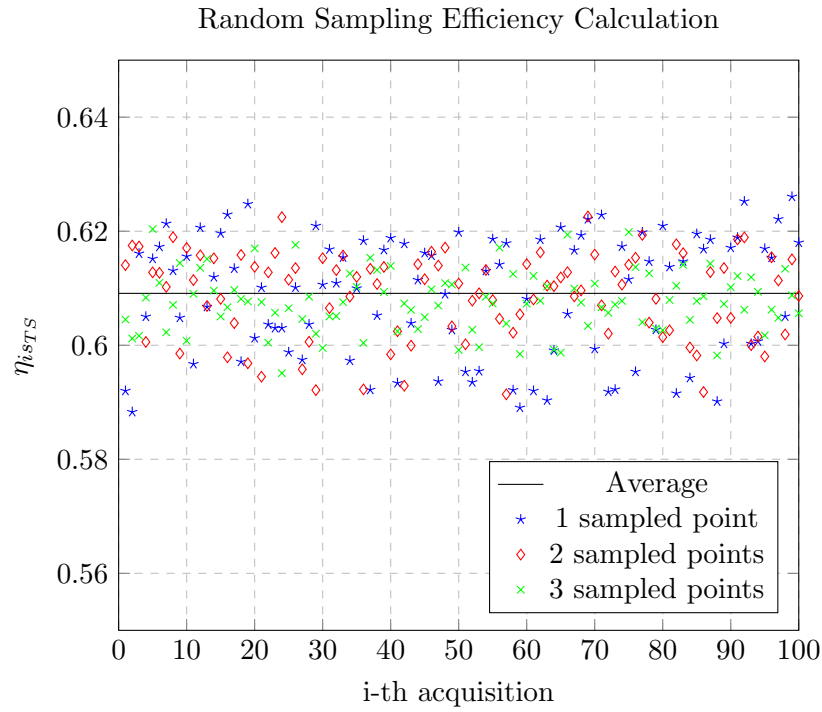


Figure 82: Efficiency, honeycomb mixer

9.3 Mixer modification

An improvement to this latter solution could be done by introducing some holes in the mixer blades. In this way, the backpressure could decrease and also a raise in the turbulence should improve the downstream mixing.

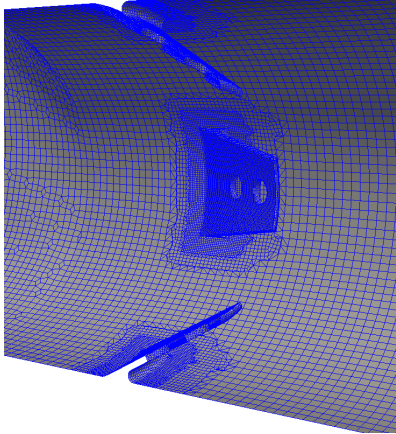


Figure 83: Grid generation on the new domain

As a matter of facts, as we could see in fig.85 and in table 38, the mixing has improved but the total pressure losses has increased as well. This is probably due to the flow constraint through the small blades holes.

In conclusion, even this solution has to be rejected.

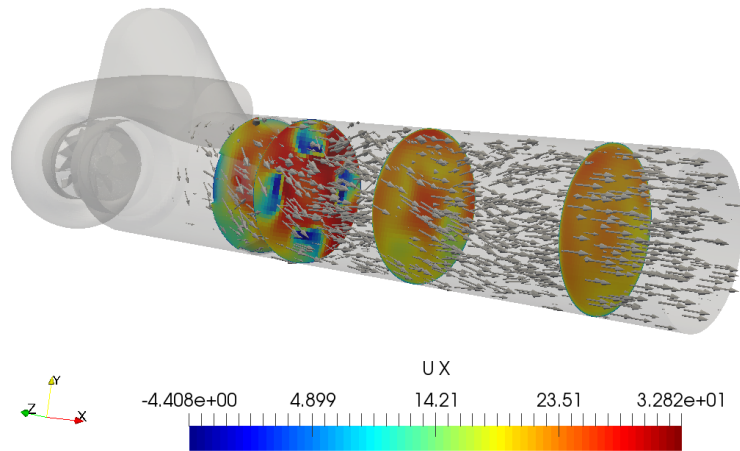


Figure 84: Graphical post-processing, modified mixer honeycomb decoupler, axial velocity contours

\mathbf{p}	\mathbf{U}_x	\mathbf{U}_y	\mathbf{U}_z	\mathbf{h}	κ	ω
$1.9e-5$	$3.0e-6$	$3.7e-6$	$2.7e-6$	$2.4e-6$	$1.2e-6$	$9.8e-8$

Table 37: Initial residuals values, honeycomb modified mixer

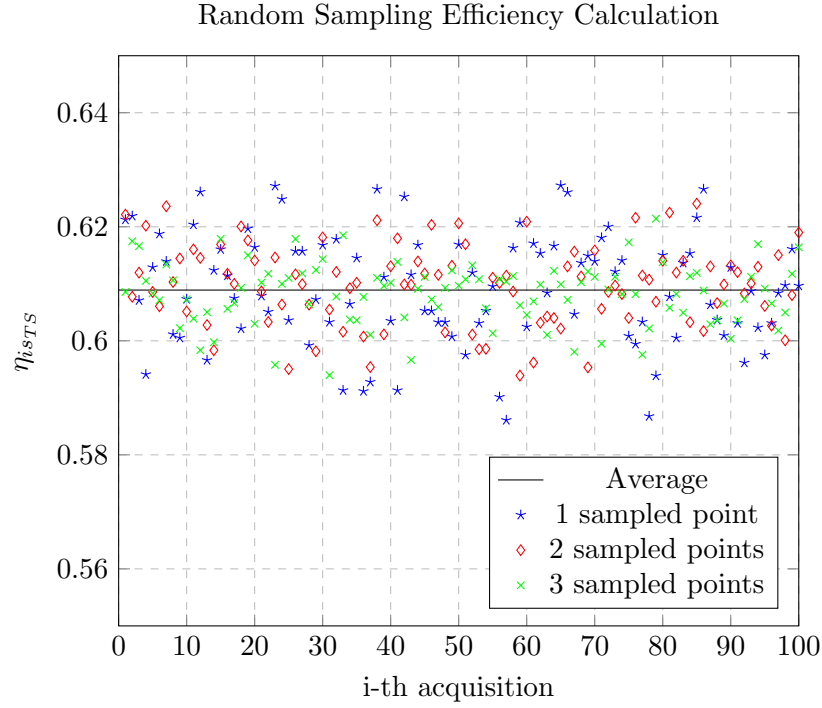


Figure 85: Efficiency, honeycomb mixer modified diffuser

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9791	1671	0.0314

Table 38: Performances indexes, honeycomb mixer

10 Pyramidal honeycomb decoupler

The solution presented in this chapter comes from the common layout of air filters cartridges for high performance internal combustion engines. Here the needs of low viscous losses is predominant. For the same reason it will be an interesting case to study.

10.1 Geometry and meshing

The idea is to build a truncated pyramidal structure where at each face corresponds an honeycomb matrix, whereas a physical wall is imposed at the top of the truncated geometry in order to force flow through the cells. Moreover, an arrow like geometry is used for the blocking wall. This will help reducing recirculation phenomena in this dead water region. Note that the casing is box shaped in order to accommodate decoupler's cartridge.

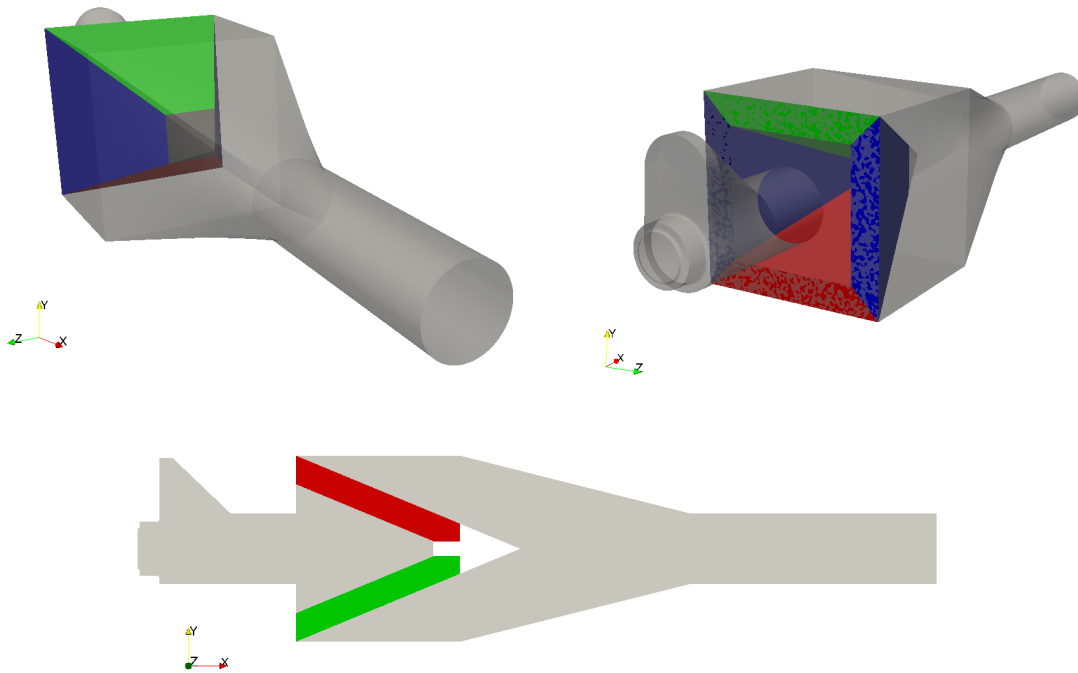


Figure 86: Front Back and section view of the generated geometry

For what concern grid generation strategy, the base parameters are kept equal with respect to previous cases. We have only added local refinement up to 1mm at the porous zones boundaries to have correctly defined cell zones. Another important aspect here stays in defining the right frame of reference for each pyramid face. Since resistance coefficients

are defined with respect to local reference, we need to insert the normal and tangential direction to the named face.

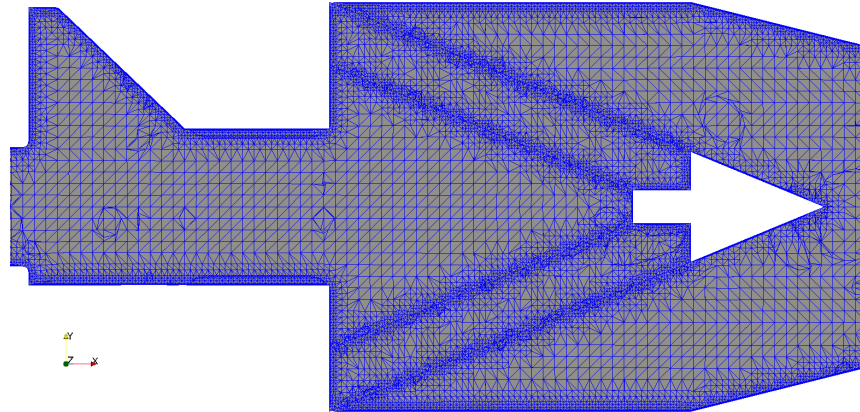


Figure 87: Grid detail, pyramidal honeycomb

10.2 Results

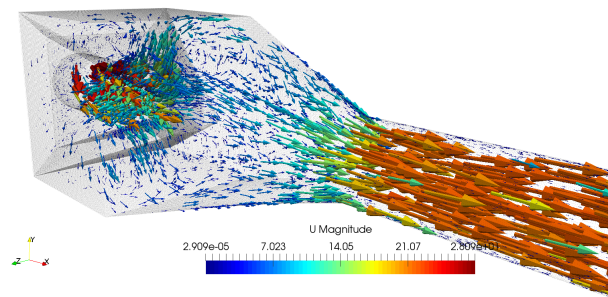


Figure 88: 3D vectors representation, velocity magnitude, pyramidal HC

\mathbf{p}	\mathbf{U}_x	\mathbf{U}_y	\mathbf{U}_z	κ	ω
$2.1e-5$	$4.4e-6$	$1.6e-5$	$1.5e-5$	$9.6e-6$	$5.3e-7$

Table 39: Initial residuals, pyramidal honeycomb

In fig.88 and 89 we could see the graphical post processing of the obtained flow fields. Note that with respect to previous cases, velocity magnitude instead of axial component

is plotted. This is due to the decoupler geometry that foresee a more three dimensional development. Thus the only axial component won't be sufficient to understand the physics.

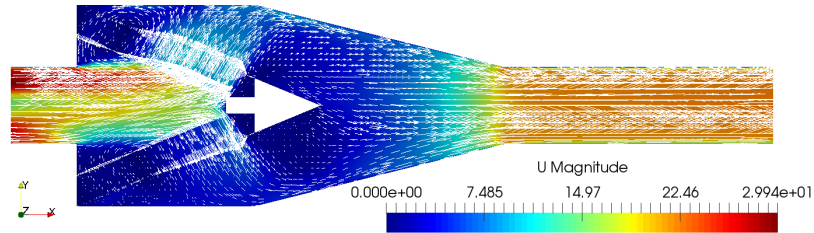


Figure 89: Axial cross section, velocity magnitude contours and vectors, pyramidal HC

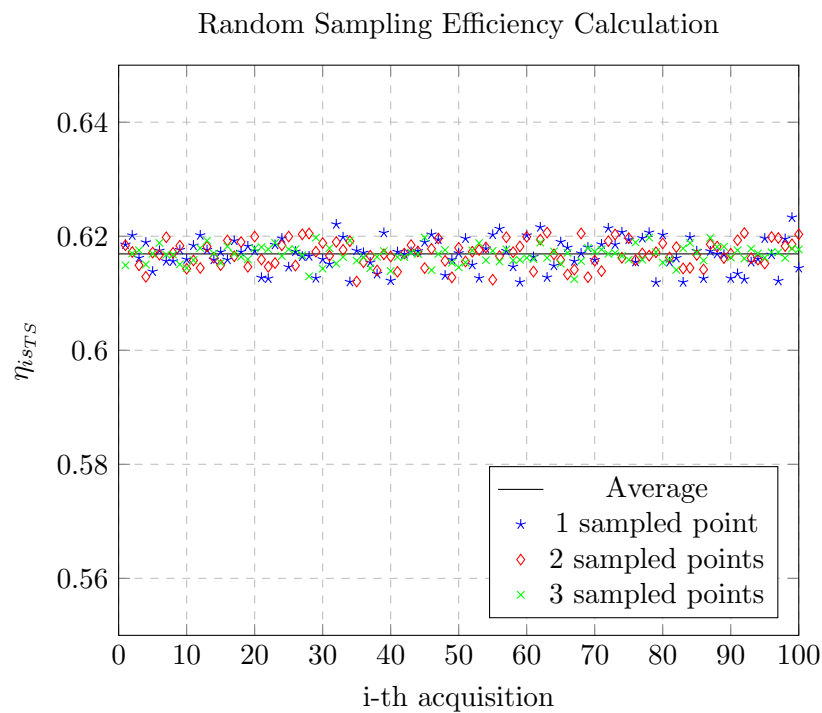


Figure 90: Efficiency, pyramidal HC decoupler

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9897	1140	0.722

Table 40: Performances indexes, pyramidal honeycomb

Fig. 90 and table 40, shows that the performances have improve both from uniformity and from losses point of view. Thus this solution will for sure deserve some deeper analysis. In fact, a new refined grid was generates, just as for the reference honeycomb expansion chamber case. This foresee an halve of the base element edge's length from 2mm to 1mm in the volute and rotor regions whereas, for what concern discharge, we have shifted from 4mm to 2mm (fig.91). Remember that when we were investigating grid independence for the base layout we have found out a relevant improve in performances.

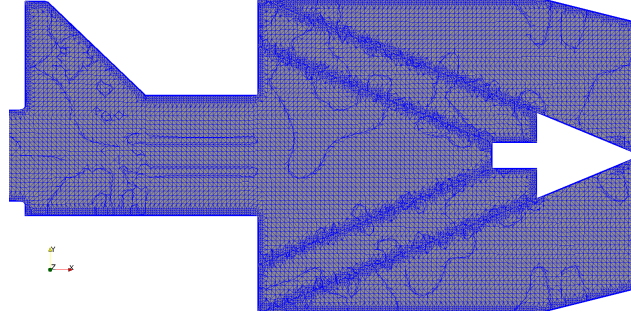


Figure 91: Grid detail, refined pyramidal honeycomb

Referring to efficiency evaluation (fig.93), as for the expansion chamber case, the value raises of almost 3%, moreover scattering around the mean value is reduced. On the other hand, pressure loss increase by 10%. However, this solution worth to be experimentally tested since it provides the best result in term of uniformity γ_m .

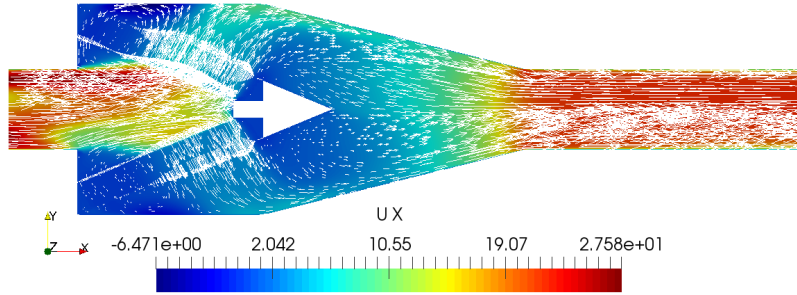


Figure 92: Axial cross section, axial velocity contours and vectors, refined pyramidal HC

p	U_x	U_y	U_z	h	κ	ω
$1.7e-5$	$1.1e-6$	$3.2e-6$	$3.9e-6$	$4.0e-6$	$2.2e-6$	$2.4e-7$

Table 41: Initial residuals values, pyramidal HC refined

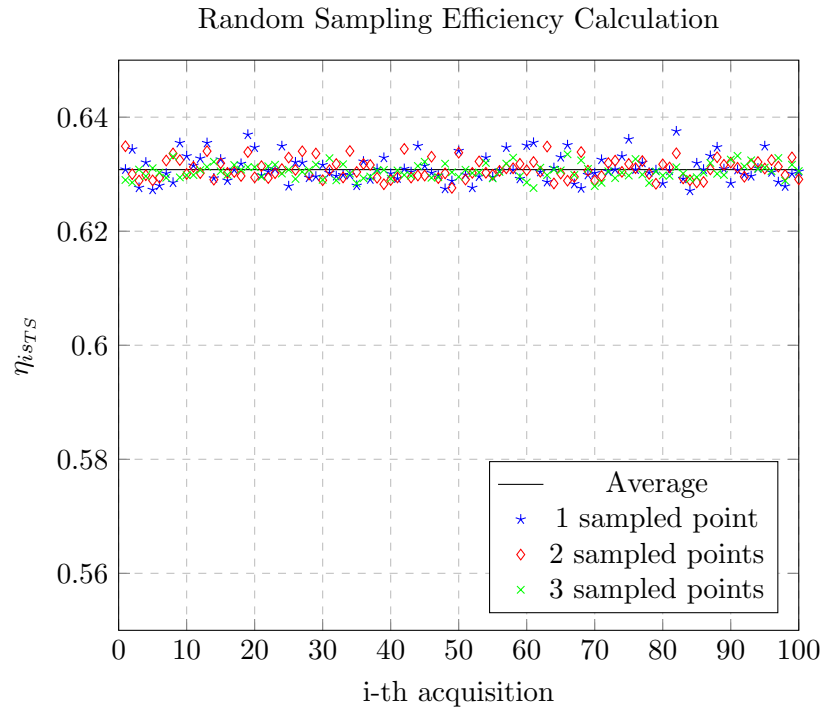


Figure 93: Efficiency, pyramidal HC decoupler, refined grid

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9905	1288	0.414

Table 42: Performances indexes, pyramidal honeycomb, refined grid

11 Hexagonal honeycomb decoupler

As a matter of facts, we have find out that using honeycombs not in the flow direction improve the performances, though we could try also different but similar layouts.

11.1 Geometry and meshing

The starting point of this solution comes from the idea of create something axisymmetric with the aim of having a uniform back-pressure on the flow. Since, usually, these elements are produced through an extrusion process it is impossible to create curved geometries. Thus an hexagon structure has been chosen. The spike has the same role as the arrow like shape in the previous geometry thus avoiding excessive dead water regions.

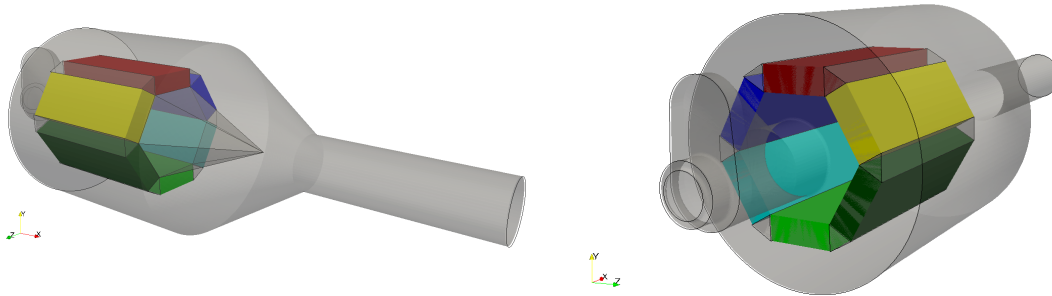


Figure 94: Front and Back view of the generated geometry

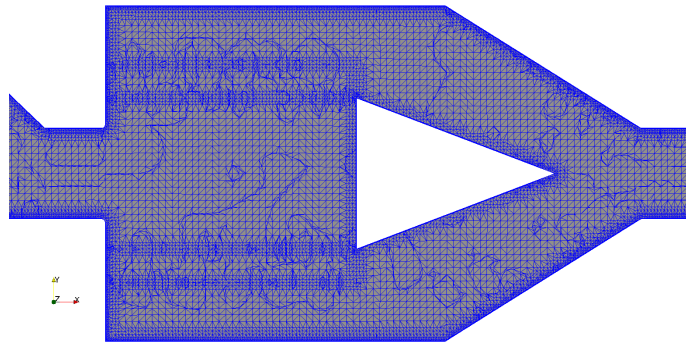


Figure 95: Grid Detail

In fig.95 it is shown the grid generated, the procedure set-up is exactly the same as for the pyramid, with base element's edges of 4mm and refinement up to 1mm at walls and cell zones boundaries.

11.2 Results

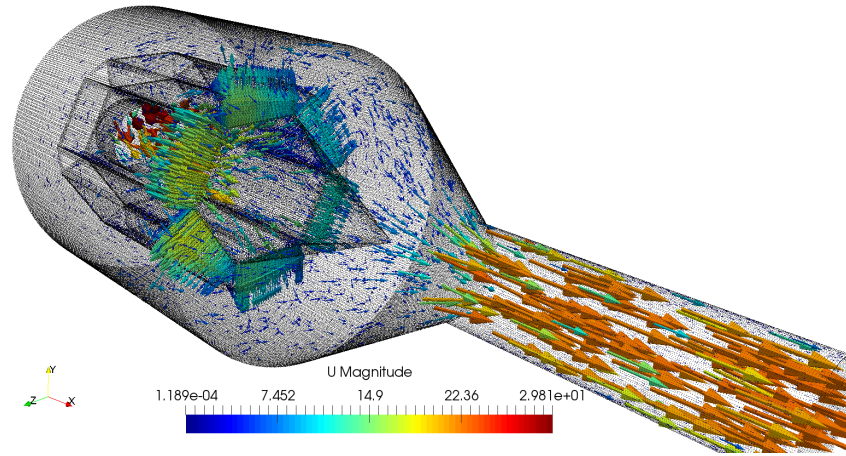


Figure 96: 3D vectors representation, velocity magnitude, hexagonal HC

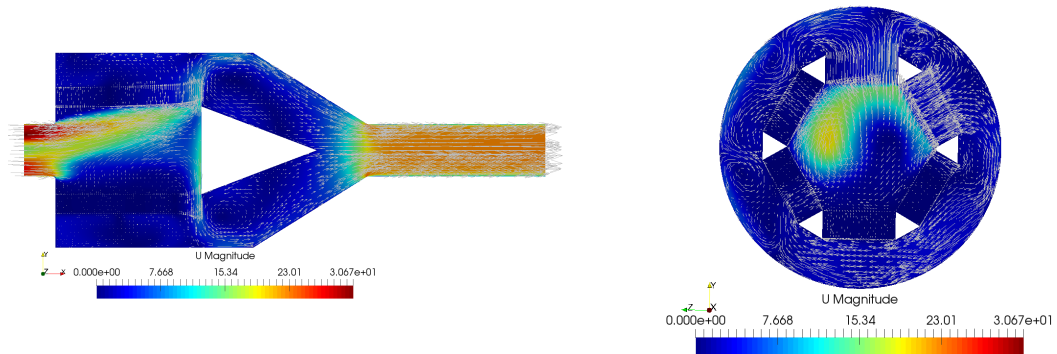


Figure 97: Multiple Sections, velocity magnitude contours and vectors, hexagonal HC

From the figures above, we could actually state that even if the geometry is axisymmetric, the flow definitely isn't. The sudden chamber expansion arise radial equilibrium issue. Moreover it is evident the fact that passage through matrices' cells is concentrated in the final portion. Looking at the sections representations we could observe large scale recirculation patterns. Though we're not in an optimal situation but lets see the performances parameters.

Grid	\mathbf{p}	\mathbf{U}_x	\mathbf{U}_y	\mathbf{U}_z	\mathbf{h}	κ	ω
Standard	$4.5e-5$	$2.8e-7$	$6.4e-6$	$4.2e-6$	$9.7e-6$	$6.7e-7$	$6.2e-8$
Refined	$5.8e-5$	$8.7e-7$	$9.1e-6$	$8.8e-6$	$1.4e-5$	$2.9e-6$	$1.1e-7$

Table 43: Initial residuals values, hexagonal HC

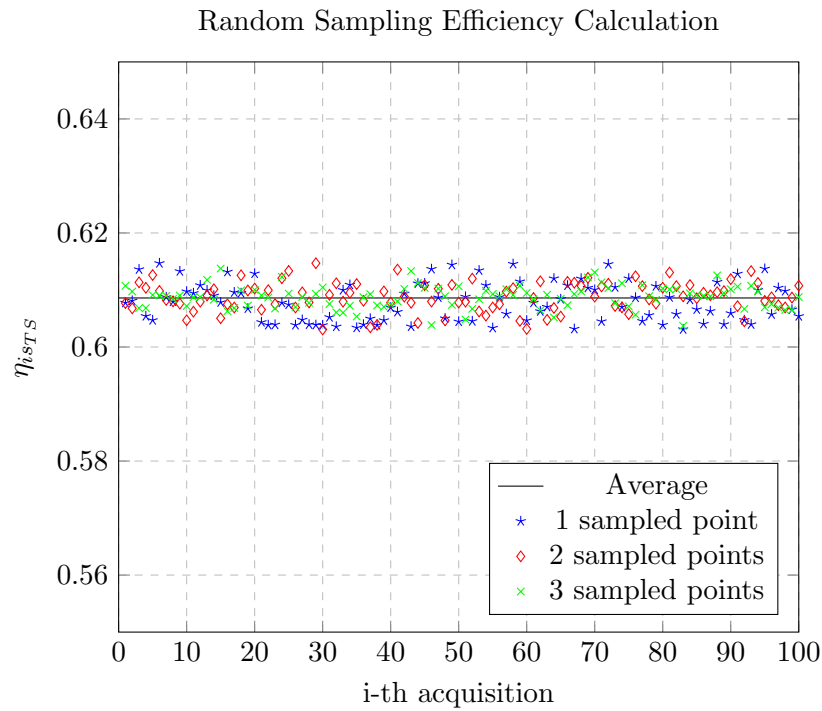


Figure 98: Efficiency, hexagonal HC decoupler, standard grid

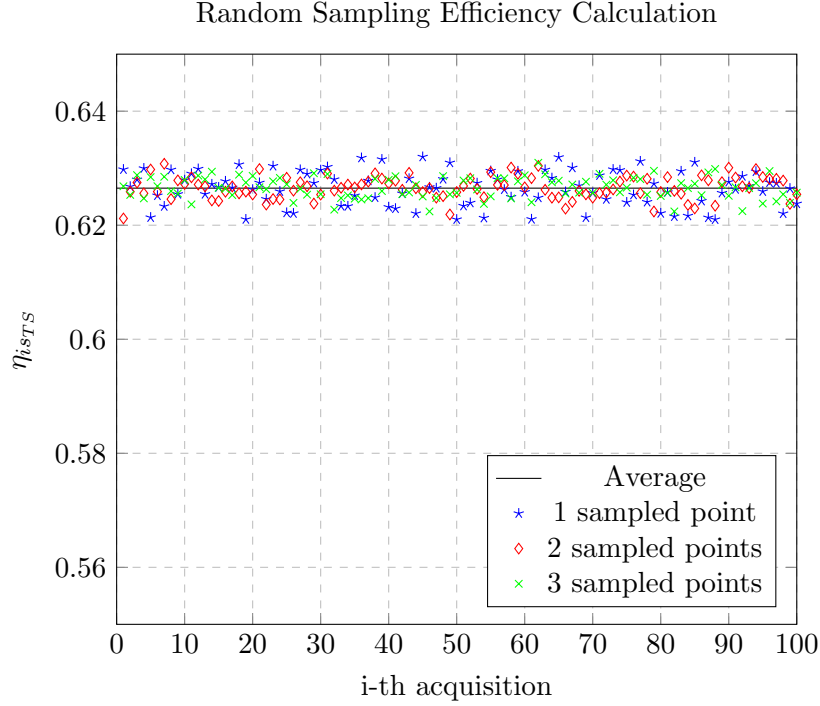


Figure 99: Efficiency, hexagonal HC decoupler, refined grid

Grid	γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
Standard	0.9898	1105	0.68
Refined	0.9889	1284	0.61

Table 44: Performances indexes, hexagonal HC decoupler

Nevertheless, performances are pretty good. Pressure loss decrease of additional 50 Pa with respect to pyramid case and meridional velocity uniformity is equal. However from the fig.98 we see that efficiency values coming from random sampling procedure are a little more spread around the mass-flow averaged value.

However, when we switch to refined grid case, we are in front of an increase in total pressure loss, whereas uniformity γ_m , remains more or less constant. As a matter of facts, at this point, results coming from pyramid and hexagonal layout are comparable. Moreover, with respect to the pyramidal case, this layout has the benefit of exploiting the original casing already physically built (expansion chamber). These considerations reveals that this device provides the most encouraging performances, thus it will also be tested in a different turbine operating point, as already done for the reference case.

11.3 Results: second operating point

As done previously we're going to shift to an upper reduced rotational speed, from 4750 to 5500 $rpm\sqrt{K}$.

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9889	1199	0.54

Table 45: Performances indexes, hexagonal HC decoupler, n_{red} 5500 $rpm\sqrt{K}$

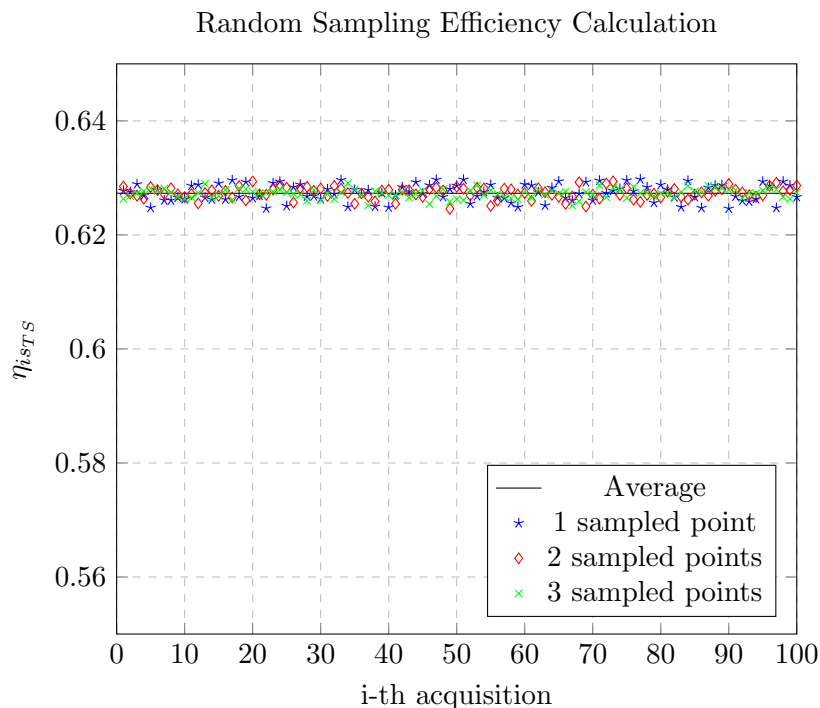


Figure 100: Efficiency, hexagonal HC decoupler, n_{red} 5500 $rpm\sqrt{K}$

Looking at the results reported above, we could state that this solution is less sensible to the variation of operating point, probably it is given by the fact of having the honeycombs placed not in the main flow direction (axial). As a consequence we have obtained very good performances both in terms of pressure loss and uniformity while maintaining the swirling velocity under unity.

11.4 Results: smaller hexagonal matrix

In this subsection we will show what happens if the decoupler matrix dimensions are reduced. This trial finds its way in the possibility of reducing the helical flow development seen with the previous layout. At this proposal the internal diameter of the device has been halved but maintaining the porous media thickness.

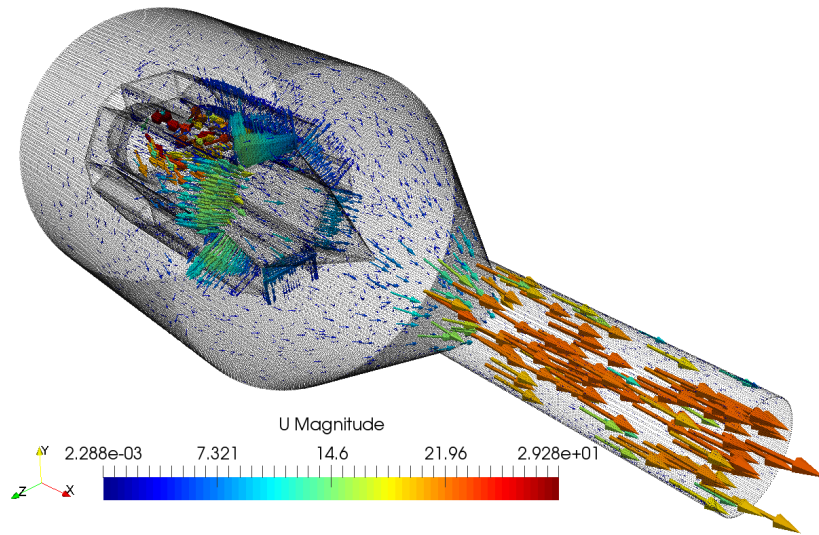


Figure 101: 3D vectors representation, velocity magnitude, small hex HC

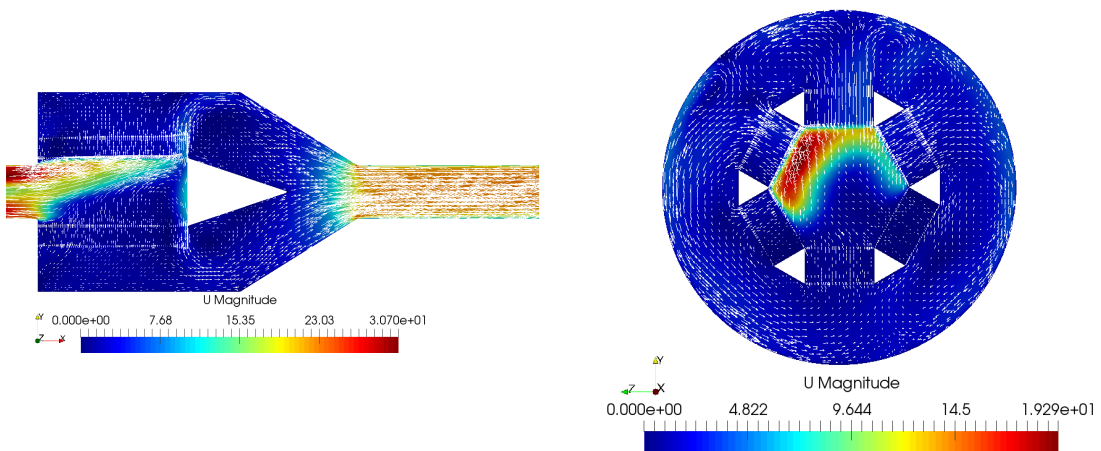


Figure 102: Multiple Sections, velocity magnitude contours and vectors, small hex HC

\mathbf{p}	\mathbf{U}_x	\mathbf{U}_y	\mathbf{U}_z	\mathbf{h}	κ	ω
$5.5e-5$	$4.7e-7$	$1.1e-5$	$6.9e-6$	$1.6e-6$	$1.3e-6$	$6.5e-8$

Table 46: Initial residuals values, smaller hex HC

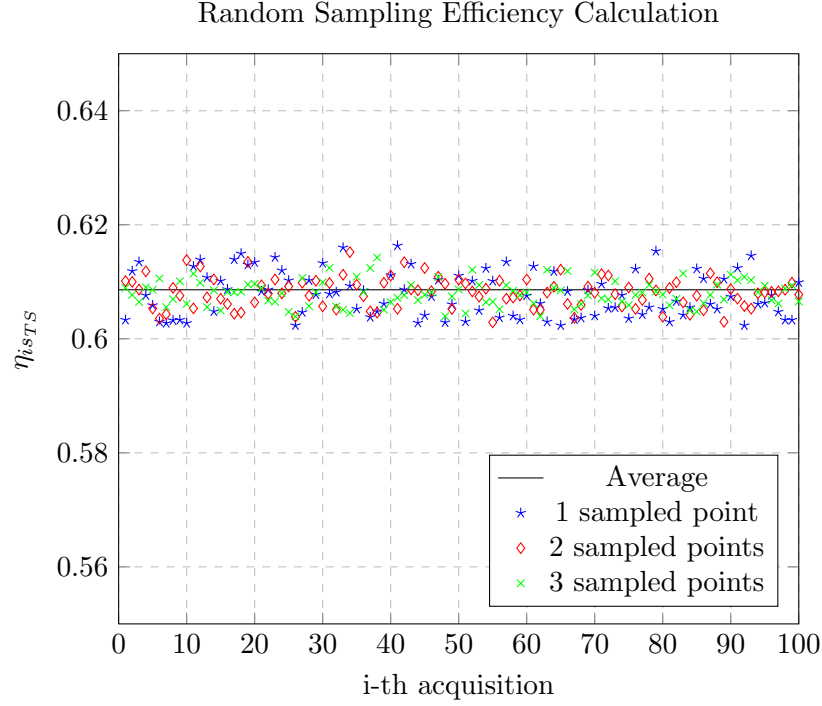


Figure 103: Efficiency, small hex HC decoupler

γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
0.9888	1106	0.5804

Table 47: Performances indexes, hexagonal honeycomb

However, results have not improved, axial velocity uniformity is slightly lower and other variables are comparable.

12 Honeycomb alternatives

Honeycomb matrices are widely spread items in automotive industry, their main advantage is the surface to volume ratio, making them particularly suited for three way catalyst applications where the chemical efficiency is directly proportional to the amount of wetted surface.

12.1 Macro Honeycomb

However, for our application we are only interested in mixing and de-swirling abilities. Since these components are a matrix of parallel axial channels they are able, as seen with all the previous results, to eliminate completely the tangential component of the flow discharged by the turbine. Here's the idea of building a sort of macro honeycomb, with the aim of reducing the pressure loss given by the narrow ducts.

12.1.1 Geometry and meshing

New grill is composed by a matrix with a given density of $1 \text{ cell}/\text{cm}^2$ along with 1 mm thick walls. The total width is of 20 mm so half the reference one. The idea is to replicate the previous numerical campaign but substituting the original 80 cpsi matrix with the new one.

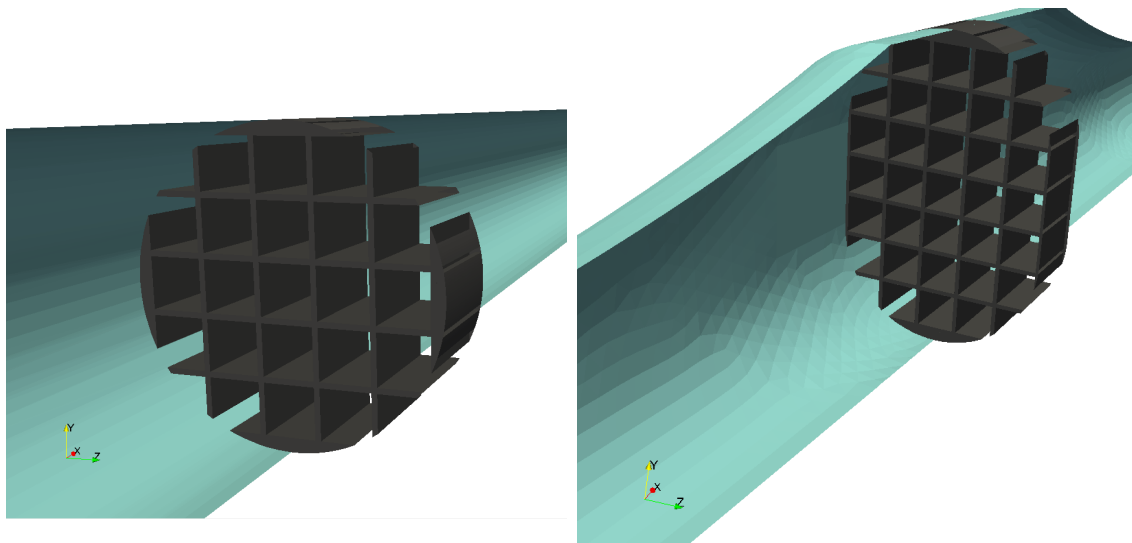


Figure 104: 3D geometry representation, standard duct and asymmetrical diffuser

Base parameters of grid generation were maintained unaltered with respect to the base cases. For ducts' walls, elements with 1 mm edge length were employed. In fig.105, we could see the results.

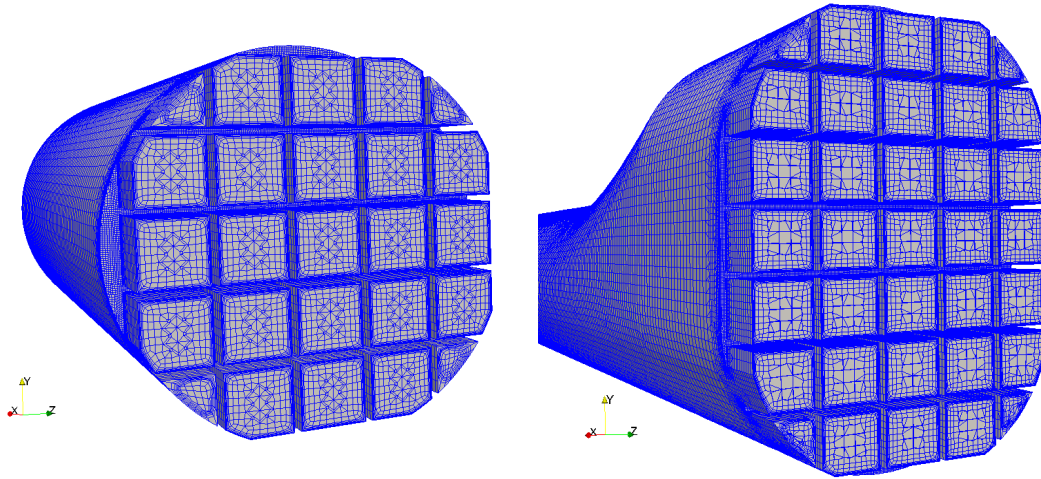


Figure 105: Grid, standard duct and asymmetrical diffuser

12.1.2 Results

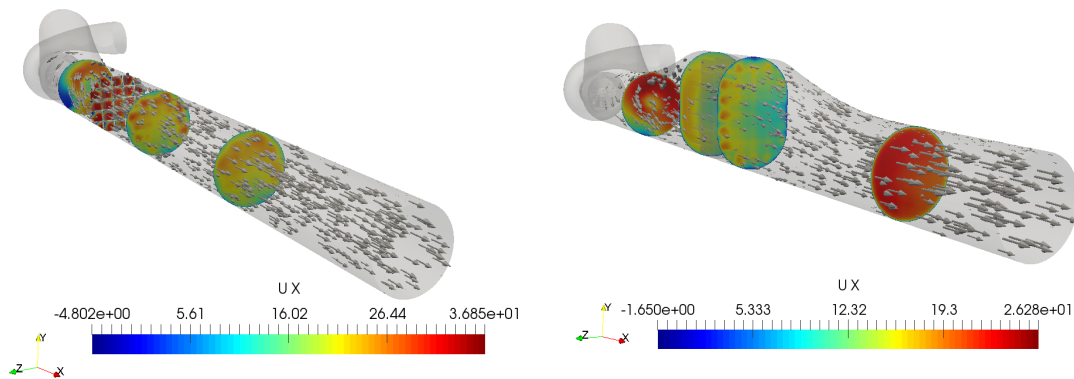


Figure 106: Grid, standard duct and asymmetrical diffuser

Geometry	\mathbf{p}	\mathbf{U}_x	\mathbf{U}_y	\mathbf{U}_z	\mathbf{h}	κ	ω
Straight	$1.7e-6$	$1.5e-6$	$9.2e-7$	$7.4e-7$	$1.4e-6$	$4.7e-7$	$4.5e-8$
Diffuser	$1.4e-6$	$3.9e-6$	$2.8e-6$	$1.9e-6$	$1.1e-5$	$7.9e-7$	$4.1e-8$

Table 48: Initial residuals values, macro HC

In table 49, we could see the performance indexes. In particular, as expected, the total pressure loss decreases with respect to the standard cases but we have to face also a worsening of uniformity. This is mainly due to the fact that mixing is penalized by the outflow from wider channels. Probably with more axial extension at our disposal, we could have achieved a good result also in this term. It is all supported also by fig. 107 and 108, that both present a more dispersed pattern with respect to the reference cases.

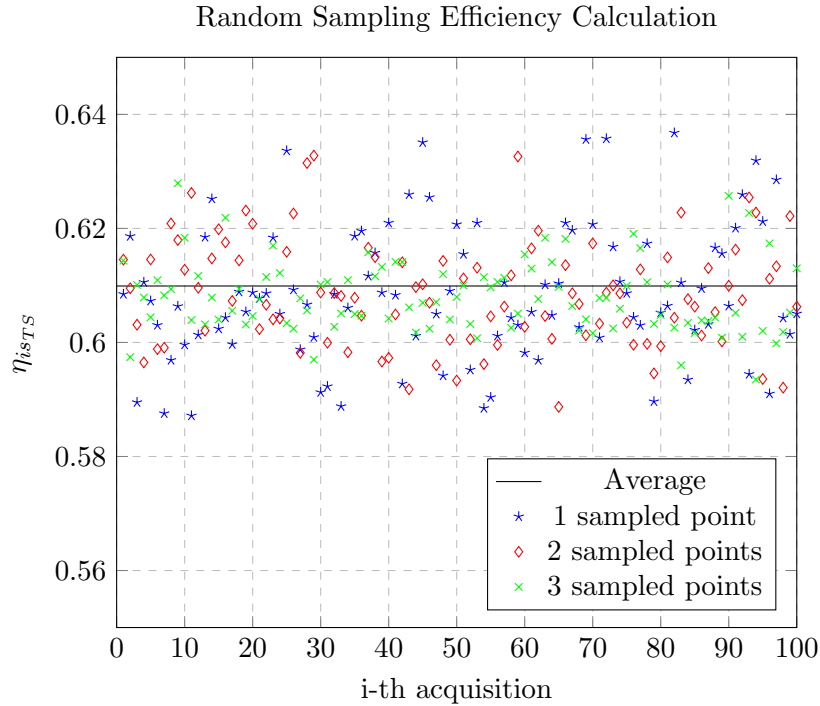


Figure 107: Efficiency, standard duct macro honeycomb

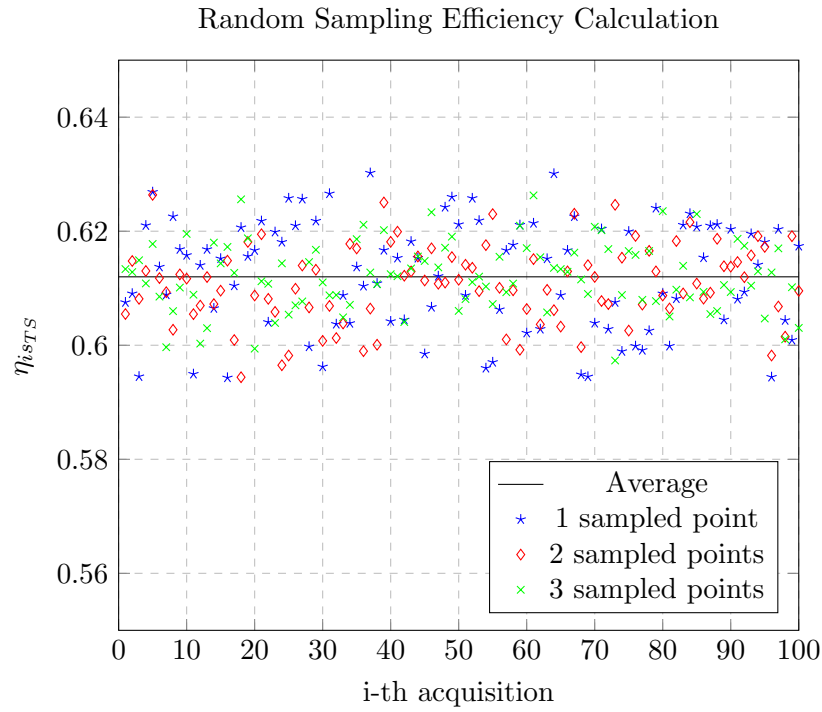


Figure 108: Efficiency, macro honeycomb diffuser

Geometry	γ_m	Δp_T [Pa]	$U_{tg,AVG}$ [m/s]
Straight	0.9660	1014	0.1170
Diffuser	0.9788	1560	0.3249

Table 49: Performances indexes, macro honeycomb

13 Final remarks

The conclusion that can be drawn from this numerical work, is that, through suitable devices, it is possible to act on the turbine discharged flow in order to make plausible an accurate direct efficiency evaluation via "standard" measuring systems and devices.

However, the issue regarding heat losses has been neglected in this investigation, mainly due to the lack of information needed for a precise modeling of such a complex phenomena. As a matter of facts, we could state that, especially in cold flow experimental campaign, it would be pretty easy to realize an almost adiabatic insulation of the decoupler devices, for example thanks to an "air-cushion" technique.

Note that the ability to treat this component as adiabatic it is a must, since we have to make sure that we do not introduce any additional heat loss with respect to the standard case.

For what concern the different devices, we have found out that homogenizing in the standard duct is pretty rough losses wise. At this point, we have to deal with expansion pressure recovery and, to do that, we must eliminate a-priori the swirling velocity component to avoid radial equilibrium related issues.

The original expansion chamber double honeycomb matrix has proven to be a very effective solution, anyhow it is very sensitive to the actual machine operating point when it comes to losses. A solution like the hexagon honeycomb matrix has proven to be a better compromise since it offers more or less constant over the turbine map while maintaining an high uniformity of discharged flow field. Now that we could consider the geometry as fixed, the future work could stay in evaluating various operating point both in terms of expansion ratio (mass flow rate) and in term of different machines. The aim will be to prove that this technique is reliable and robust for the whole radial in-flow turbines environment.

14 Bibliography

References

- [1] Borgnakke C., Sonntag, R.E., (2009). Fundamentals of Thermodynamics, seventh edition.
- [2] Whitfield A., Baines N.C., Design of Radial Turbomachines.
- [3] Shaaban S., Seume JR, 2006, Analysis of turbocharger non-adiabatic performance, 8.th International Conference on Turbochargers and Turbocharging
- [4] R.D.Burke, P. Olmeda, F. J. Arnau and M. Reyes-Belmonte, Modeling of turbocharger heat transfer under stationary and transient engine operating conditions
- [5] Serrano J. R., Olmeda P., Pez A, Vidal F., 2010, An experimental procedure to determine heat transfer properties of turbochargers, IOP publishing.
- [6] Roger Temam(1984), NavierStokes Equations: Theory and Numerical Analysis, ACM Chelsea Publishing
- [7] Montenegro G., notes from the course of Modeling Techniques for Fluid Machines, Politecnico di Milano
- [8] Persico G., slides from the course of Modeling Techniques for Fluid Machines, Politecnico di Milano
- [9] Menter, F. R. 1994, Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications, AIAA Journal
- [10] Dr. Franjo Jureti, cfMesh User Guide, Creative Fields Ltd.
- [11] V. A. Jambhekar, Forchheimer Porous-media Flow Models, Universitt Stuttgart
- [12] openFoamWiki, MRF equations development
- [13] Christopher J. Greenshields, openFoam User Guide 4.0
- [14] Singh S.N., Agrawal D.P. , Sapre R.N., Malhotra R.C., Effect of inlet swirl on the performance of wide-angled annular diffusers.
- [15] Christopher J. Greenshields, *OpenFoam Guide*, OpenFoam Direct Ltd.
- [16] Ferziger J H, Peric M, Computational Method for Fluid Dynamic.
- [17] *www.ichrome.com*, Turbulence Calculator.

A Case Setup Dictionaries

Listing 3: thermophysicalProperties

```
thermoType
{
type          hePsiThermo;
mixture       pureMixture;
transport     sutherland;
thermo        janaf;
equationOfState perfectGas;
specie        specie;
energy        sensibleEnthalpy;
}

mixture
{
specie
{
nMoles        1;
molWeight     28.96;
}
thermodynamics
{
Tlow          200;
Thigh         6000;
Tcommon       1000;
highCpCoeffs (...);
lowCpCoeffs   (...);
}
transport
{
As            1.4584e-06;
Ts            110.4;
}
}
```

Listing 4: cfMesh dictionary

```
surfaceFile "rotor.fms";
maxCellSize 2e-3; //[m]

localRefinement
{
  rotor
  {
    cellSize 5e-4;
  }
  rotorWall
  {
    cellSize 5e-4;
  }
  rotorCoupling
  {
    cellSize 5e-4;
  }
  rotorOutlet
  {
    cellSize 1e-3;
  }
}

edgeMeshRefinement
{
  root
  {
    edgeFile "edgeRef.vtk";
    cellSize 1.5e-4;
    refinementThickness 0.00025;
  }
}

surfaceMeshRefinement
{
  leadingedges
  {
    surfaceFile "refSurfaces.stl";
    cellSize 1.5e-4;
    refinementThickness 0.00025;
  }
}

boundaryLayers
{
  patchBoundaryLayers
  {
    rotorWall
    {
```

```

nLayers          3;
thicknessRatio   1.4;
maxFirstLayerThickness 1.8e-4;
allowDiscontinuity 1;
}

rotor
{
nLayers          3;
thicknessRatio   1.4;
maxFirstLayerThickness 7.2e-5;
allowDiscontinuity 1;
}
}

optimiseLayer 1;
}

```

Listing 5: topoSetDict rotor

```

...
{
name    rotor;
type    cellSet;
action  new;
source  boxToCell;
sourceInfo
{
box (-1 -1 -1) (1 1 1);
}
}
...

```

Listing 6: topoSetDict porous

```

{
name    porous1;
type    cellSet;
action  new;
source  cylinderToCell;
sourceInfo
{
p1      (0.260 0 0); // start point on cylinder axis
p2      (0.300 0 0); // end point on cylinder axis
radius  0.120;
}
}

{
name    porous2;
type    cellSet;
}

```

```

action new;
source cylinderToCell;
sourceInfo
{
p1      (0.185 0 0); // start point on cylinder axis
p2      (0.225 0 0); // end point on cylinder axis
radius  0.120;
}
}

```

Listing 7: createPatchDict

```

...
// AMI: volute-rotor
{
name voluteAMI;
patchInfo
{
type cyclicAMI;
neighbourPatch rotorInletAMI;
transform noOrdering;
}
constructFrom patches;
patches (voluteCoupling);
}
{
name rotorInletAMI;
patchInfo
{
type cyclicAMI;
neighbourPatch voluteAMI;
transform noOrdering;
}
constructFrom patches;
patches (rotorCoupling);
}
...

```

Listing 8: AMI weights result

AMI: Patch target sum(weights) min/max/average = 0.985417, 1, 0.999866

B Boundary Conditions

B.1 MRFrhoSimpleFoam

Listing 9: temperature boundary condition

```
...
inlet
{
type          totalTemperature;
gamma         1.399;
T0            uniform 353.15;
}
outlet
{
type          zeroGradient;
}
dischargeWall //adiabatic case
{
type          zeroGradient;
}
dischargeWall //fixed wall temperature case
{
type          fixedValue;
value         uniform 330;
}
".*Wall" //imposed wall heat flux case
{
type          wallHeatTransfer;
Tinf          uniform 298; //ambient temperature
alphaWall    uniform 1475; //heattransfer coefficient W/m^2
value        uniform 299; //dummy entry
}
...

```

Listing 10: pressure boundary conditions

```
...
inlet // mass flow rate BC
{
type          zeroGradient;
}
inlet // total to static pressure ratio BC
{
type          totalPressure;
rho           none;
psi           thermo:psi;
gamma         1.399;
p0            143150;
}

```

```

value          $internalField;
}
outlet
{
type          fixedMean;
meanValue     uniform 102250;
value         $internalField;
}
...

```

Listing 11: velocity boundary conditions

```

...
inlet // mass flow rate BC
{
type          flowRateInletVelocity;
massFlowRate  0.050;
rho           rho;
rhoInlet      1;
}
inlet // total to static pressure ratio BC
{
type          pressureInletVelocity;
value         uniform (0 0 1);
}
outlet
{
type          zeroGradient;
}
...

```

Listing 12: turbulent kinetic energy BC

```

...
inlet
{
type          turbulentIntensityKineticEnergyInlet;
intensity     0.05;
value         $internalField;
}
...

```

Listing 13: turbulent specific frequency BC

```

...
inlet
{
type          turbulentMixingLengthFrequencyInlet;
mixingLength  0.01;
value         $internalField;
}

```

```
}  
...
```

Listing 14: wallFunctions

```
".*Wall"  
{  
type          omegaWallFunction;  
value         $internalField;  
}  
  
".*Wall"  
{  
type          kqRWallFunction;  
value         $internalField;  
}  
  
".*Wall"  
{  
type          compressible::alphatWallFunction;  
value         uniform 0;  
}  
  
".*Wall"  
{  
type          nutkWallFunction;  
value         uniform 0;  
}
```

Listing 15: MRFproperties

```
...  
{  
cellZone      rotor;  
active        yes;  
  
nonRotatingPatches ( voluteAMI rotorInletAMI rotorOutletAMI  
dischargeInletAMI shroudWall backWall );  
  
origin        (0 0 0);  
axis          (1 0 0);  
omega        11932;  
}  
...
```


B.2 Porous flow-metering

Listing 16: temperature boundary conditions

```
internalField    uniform 573;

boundaryField
{
    ".*Wall"
    {
        type      zeroGradient;
    }

    inlet
    {
        type      totalTemperature;
        gamma     1.365;
        psi       thermo:psi;
        T0        uniform 573;
        value     $internalField;
    }
    outlet
    {
        type      zeroGradient;
    }
}
```

Listing 17: pressure boundary conditions

```
internalField    uniform 101325;

boundaryField
{
    ".*Wall"
    {
        type      zeroGradient;
    }

    inlet
    {
        type      zeroGradient;
    }
    outlet
    {
        type      fixedMean;
        meanValue 101325;
        value     uniform 101325;
    }
}
```

Listing 18: velocity boundary conditions

```
internalField    uniform (1 0 0);

boundaryField
{
    ".*Wall"
    {
        type            noSlip;
    }

    inlet
    {
        type            flowRateInletVelocity;
        massFlowRate    0.03078;
        rho              rho;
        rhoInlet         0.625;
    }
    outlet
    {
        type            inletOutlet;
        inletValue       uniform (0 0 0);
        value            uniform (0 0 0);
    }
}
}
```

B.3 Porous turbulence evaluation

Listing 19: boundary conditions

```
..VelocityBC..
inlet
{
    type            fixedValue;
    value           uniform (8 0 0);
}

outlet
{
    type            zeroGradient;
}
...

..PressureBC..
inlet
{
    type            zeroGradient;
}
outlet
{
    type            fixedMean;
}
```

```

        meanValue      101325;
        value          uniform 101325;
    }
..
..kBC..
    inlet
    {
        type          turbulentIntensityKineticEnergyInlet;
        intensity     0.05;
        value         $internalField;
    }

    outlet
    {
        type          zeroGradient;
    }
...
..omegaBC..
    inlet
    {
        type          turbulentMixingLengthFrequencyInlet;
        mixingLength  0.001;
        value         $internalField;
    }

    outlet
    {
        type          zeroGradient;
    }
..

```

B.4 fvOptions porous media

Listing 20: fvOptions

```

porosity1
{
    type          explicitPorositySource;
    active       yes;

    explicitPorositySourceCoeffs
    {
        selectionMode  cellZone;
        cellZone      porousHC;

        type          DarcyForchheimer;

        DarcyForchheimerCoeffs
        {
            d          (2e6 -1 -1);
        }
    }
}

```

```

        f      (10 0 0);

        coordinateSystem // Cartesian coordinates for the cellZone
        {
            x      (1 0 0);
            y      (0 1 0);
            #includeEtc "caseDicts/general/coordinateSystem/cartesianXY"
        }
    }
}

porosityTubulenceSink
{
    type          scalarSemiImplicitSource;
    active        yes;

    scalarSemiImplicitSourceCoeffs
    {
        selectionMode    cellZone;
        cellZone          porousHC;

        volumeMode        specific; //absolute; //
        injectionRateSuSp
        {
            omega        (1e5 0);
            k             (0 -1e3);
        }
    }
}

```

C Numerical Schemes

C.1 MRFrhoSimpleFoam

Listing 21: fvSchemes

```

ddtSchemes
{
    default          steadyState;
}

gradSchemes
{
    default          Gauss linear;
}

divSchemes
{
    default          none;
    div(phi,U) bounded Gauss upwind;
}

```

```

div(phi,k) bounded Gauss upwind;
div(phi,omega) bounded Gauss upwind;
div(phi,epsilon) bounded Gauss upwind;
div(phi,nuTilda) bounded Gauss upwind;
div(phi,K) bounded Gauss upwind;
div(phi,h) bounded Gauss upwind;
div(phiid,p) bounded Gauss upwind;
div((p*(U-URel))) Gauss linear;
div((-devRhoReff.T()&U)) Gauss linear;
div(((rho*nuEff)*dev2(T(grad(U)))) Gauss linear;
}
laplacianSchemes
{
default          Gauss linear corrected;
}
interpolationSchemes
{
default          linear;
}
snGradSchemes
{
default          corrected;
}
fluxRequired
{
default no;
}
p ;
}
wallDist
{
method          meshWave;
}

```

Listing 22: fvSolution

```

solvers
{
p
{
solver GAMG;
tolerance 1e-07;
relTol 0.01;
smoother GaussSeidel;
cacheAgglomeration true;
agglomerator faceAreaPair;
}
U
{
solver smoothSolver;
smoother GaussSeidel;
tolerance 1e-7;
}
}

```

```

relTol 0.1;
}
"(rho|h|k|epsilon|omega)"
{
solver          smoothSolver;
smoother        symGaussSeidel;
tolerance       1e-20;
relTol          0.1;
}
}

SIMPLE
{
nNonOrthogonalCorrectors 2;
rhoMin 0.1;
rhoMax 20;
transonic      false;
consistent     no;
}
relaxationFactors
{
fields
{
p          0.3;
rho       0.2;
}
equations
{
U          0.7;
U          0.5;
"(k|epsilon|omega)" 0.4;
}
}
}

```

C.2 Porous turbulence evaluation

Listing 23: fvSchemes

```

ddtSchemes
{
default          steadyState;
}

gradSchemes
{
default          Gauss linear;
}

divSchemes
{
default          none;
div(phi,U)      limitedLinearV 1;/
}

```

```

    div(phi,k)      bounded Gauss upwind;
    div(phi,omega) bounded Gauss upwind;
    div((nuEff*dev2(T(grad(U)))) Gauss linear;
}
laplacianSchemes
{
default          Gauss linear corrected;
}
interpolationSchemes
{
default          linear;
}
snGradSchemes
{
default          corrected;
}
fluxRequired
{
default no;
p ;
}
wallDist
{
method          meshWave;
}

```