

Numerical study of flow in diffuser

Investigation with three turbulence models and comparison with experimental data

Bachelor's thesis

JOHAN ROGESTEDT, MARTIN FORSELL, MOHAMED ZABEN

Department of Mechanics and Maritime Sciences
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2018

BACHELOR'S THESIS FOR MECHANICS AND MARITIME SCIENCES
2018:02

Numerical study of flow in diffuser

Investigation with three turbulence models and comparison with
experimental data

JOHAN ROGESTEDT, MARTIN FORSELL, MOHAMED ZABEN



CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Mechanics and Maritime Sciences
Division of Fluid Dynamics
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2018

© Johan Rogestedt 2018.
© Martin Forsell 2018.
© Mohamed Zaben 2018.

Supervisor: Dr. Ardalan Javadi, Department of Mechanics and Maritime Science
Examiner: Prof. Håkan Nilsson, Department of Mechanics and Maritime Science

Bachelor's thesis 2018:02
Department of Mechanics and Maritime Science
Division of Fluid Dynamics
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: Velocity profile for flow entering a diffuser after separation has occurred due to blockage in the form of a center body. In practical applications this center body before the diffuser part could be a bearing hub.

Typeset in L^AT_EX
Gothenburg, Sweden 2018

Abstract

Fluid dynamics is a branch of physics that has many applications when it comes to solving real-world problems. Areas that utilise the knowledge in fluid dynamics in order to develop technology that is more efficient and less resource craving are diverse and range from the energy sector to the transport sector. Furthermore, as computational power increases, it becomes both more cost- and time effective to move from conducting practical experiments to instead perform numerical computer simulations. This project has studied the possibility of conducting numerical simulations of the separation phenomenon and also if it is possible to capture the effects of flow control meant to minimise separation. More specifically this thesis has focused on the case with flow through a conical diffuser with an annular inlet and a center body present, where the center body causes the flow to separate. This center body before the diffuser part could in practical applications be in the form of a bearing hub. Separation is usually an undesirable feature for flows in confined space and the ability to counteract its development through both passive (geometrical alterations) and active (for example injection of jets) flow control mechanisms are important tools to an optimised diffuser design. The availability of the well performing open source program such as OpenFOAM is a further reason to why the development of accurate numerical methods is of particular interest.

The simulations undertaken in this project has produced results in the form of three component mean velocity distributions of the flow in a conical diffuser with an annular inlet and center body present. Results were then compared with experimental reference data. The investigation covered three turbulence models ($k-\omega$, $k-\omega$ -SST and $k-\omega$ -SST-SAS) in two different geometries, corresponding to the implementation of a passive separation control feature by implementing a straight section after the center body (in order to minimise separation and in accordance with reference experiment for comparison). Additionally, the effect of a flow control method called Coanda blowing in order to minimise separation was thoroughly investigated. A swirl component of the inlet velocity in order to increase pressure recovery was another implementation made and compared to reference data.

The results showed that the $k-\omega$ and $k-\omega$ -SST models managed to capture the general motion of the flow with and without passive flow control. However, the Coanda effect from the jets proved difficult to capture, even when a refined mesh was created. The $k-\omega$ -SST-SAS model proved ineffective here, despite its said superiority in previous experiments. Probably, lack of previous experience in CFD simulations as well as higher mesh demands for the SAS model explains its poor results. The results, however, validated the proposed coupling between separation in the wake and in the wall boundaries as well as how swirl can generate an increased pressure recovery.

Keywords: separation, Coanda effect, diffuser, OpenFOAM, $k-\omega$, SST, SAS.

Acknowledgements

This project was done during the autumn 2017 at Chalmers University of Technology. The participants in this thesis are students of Master of Science in Engineering programs, including both from Mechanical Engineering and Engineering Physics.

We will start by thanking our supervisor Ardalan Javadi for the guidance and the constructive opinions and remarks we have received during the project. We would also like to thank our examiner, Prof. Håkan Nilsson for the opportunity to perform this project as well as his assistance with providing computational time. Lastly, we would like to thank Bercelay Niebles Atencio for further help with the mesh generation.



Contents

List of Figures	xi
List of Tables	xv
1 Introduction	1
1.1 Aim	2
1.2 Limitations	3
2 Theory	5
2.1 Navier Stokes equations to URANS	5
2.2 Turbulence modelling	6
2.2.1 RAS turbulence models	7
2.2.1.1 k - ω model	8
2.2.1.2 $k - \omega$ -SST model	9
2.2.1.3 k - ω -SST-SAS	10
2.3 Discretisation	10
2.3.1 Geometric discretisation	11
2.3.2 Equation discretisation	11
2.3.3 Solver selection	12
2.3.3.1 PIMPLE	12
2.3.4 Courant condition	12
2.4 Characteristics in boundary layer	13
2.5 The separation phenomenon	14
2.5.1 Separation	15
2.5.2 Coanda effect	16
3 Method	17
3.1 Generation of mesh	17
3.1.1 Geometrical setups	17
3.1.1.1 Geometric setup without straight section	18
3.1.1.2 Geometric setup with straight section	20
3.1.2 Mesh quality and y^+	21
3.1.3 Implementation of flow control	22
3.1.3.1 Coanda blow as an active flow control	23
3.1.3.2 Induced swirl as an active flow control	23
3.2 Simulation using OpenFOAM	25

3.2.1	Choice and implementation of solver	25
3.2.1.1	Values of initial conditions	25
3.2.2	Strengths and weaknesses of the implemented turbulence models	25
3.3	Post-processing	26
4	Results	27
4.1	Case I, II & III	28
4.2	Case IV, V & VI	28
4.3	Case VII and VIII	33
4.4	Case X, XI	36
4.5	Case XIII	36
5	Discussion	43
5.1	Comparison between simulations and experimental data without Coanda jet	43
5.1.1	Cases I, II & III (no straight section)	43
5.1.2	Cases VII, VIII, IX (with straight section)	45
5.2	Discussion of results from simulations with Coanda blowing	47
5.3	Discussion regarding the results from the SAS-simulations	49
5.4	Effects on results from improvement of mesh	50
5.4.1	Comparison between old and improved mesh	50
5.4.2	The effect of increased jet velocity	50
5.5	The effect of swirl on pressure recovery	52
6	Conclusion	55
6.1	Future Work	56
	Bibliography	57
A	Appendix	I
A.1	y-Plus values	I
A.2	Boundary values	I
A.3	OpenFOAM code for swirl implementation	II
A.4	Implementation in OpenFOAM	VIII
A.4.1	Implementation of turbulence models	VIII
A.4.1.1	$k-\omega$ and $k-\omega$ -SST	VIII
A.4.1.2	$k-\omega$ -SST SAS	VIII
A.4.2	Other noteworthy changes to the OpenFOAM settings for the case	IX

List of Figures

2.1	The buildup of different regions of the boundary layer near a wall. (Credits for the image goes to L. Davidson [6]. Used with permission.)	13
2.2	Graph over the flow behaviour near the wall where the regions for where the viscous layer model (the dashed black line) and the logarithmic law model (the dashed red line) are valid can easily be seen. The blue line is data from a DNS (Direct Numerical Simulation). (Credits for the image goes to L. Davidson [6]. Used with permission.)	14
3.1	A detailed description of the measurements used for the center body and the surrounding outer wall section. Amongst other, a noteworthy feature is the fact that the coordinate system used in this work as well as in the experimental project has its origin at the nose of the center body. Credits for the image goes to Eaton and Lo [2]. Used with permission.	19
3.2	A figure to illustrate how the Coanda jets were implemented in the experimental setup. This setup was mimicked in a numerical sense by the creation of a separate geometric pieces representing the jets, where furthermore the effect of the jets could be implemented through a boundary condition corresponding to the blow rate wanted. Credits for the image goes to Eaton and Lo [2]. Used with permission.	20
3.3	A visualisation of the center body used in the experiment. Relevant parts to notice are the O-ring (green) and the Coanda tailpiece (yellow).	20
3.4	A view of the second geometry generated in ICEM CFD, with an elongated cylindrical section, seen from the side. The measurements can be found listed in the text.	21
3.5	A schematic view of the geometry with the straight section implemented after the center body. Credits for the image goes to Eaton and Lo [2]. Used with permission.	21
3.6	A visualisation of the mesh used for $k-\omega$ and $k-\omega$ -SST simulation. The mesh contains approximately 250000 cells and a y^+ at least greater than 20 globally and locally fulfills $30 < y^+ < 100$ locally. The shot is taken with the "plane function", which failed to give a complete view of the cells close to the center body wall.	23

3.7	A visualisation of the mesh used for $k - \omega$ -SST-SAS simulation. This mesh has about 1800000 cells, making it roughly six times as many as the one used for $k - \omega$ and $k - \omega$ -SST. Noteworthy is how this mesh has a significantly higher concentration of cells close to the walls than the one in figure 3.6, as well as the fact that the number of cells in the area right after the end of the center body is very high compared to other areas of the same mesh.	24
3.8	A visualisation of how the Coanda effect is implemented as an active flow control to achieve control over the separating wake. The arrows shows how the injected water, by jets surrounding the end piece, follows curvature of the end and consequently removes the wake. Credits for the image goes to Eaton and Lo [2]. Used with permission	24
4.1	Top: a numerical representation of the velocity distribution with $k - \omega$ modelling (Case I). Bottom: plot showing the residuals of U_x , implying that the results have converged.	29
4.2	Top: a numerical representation of the effect of velocity distribution with $k - \omega$ -SST modelling(Case II). Bottom: plot showing the residuals of U_x , implying that the results have converged.	30
4.3	Top: a numerical representation of the velocity distribution with $k - \omega$ -SST-SAS modelling(Case III). Bottom: a plot showing the residuals of U_x , implying that the results have converged.	31
4.4	Top: a numerical representation of the velocity distribution under the effect from a Coanda jet with $k - \omega$ modelling (Case IV). Bottom: plot showing the residuals of U_x , implying that the results have converged.(final residual=6.98e-6)	32
4.5	Top: a numerical representation of the velocity distribution under the effect from a Coanda jet with $k - \omega$ -SST modelling (Case V). Bottom: a plot showing the residuals of U_x , implying that the results have converged.(Final residual = 8.78e-6)	33
4.6	Top: a numerical representation of the velocity distribution under the effect from a Coanda jet with $k - \omega$ -SST-SAS modelling (Case VI). Bottom: a plot showing the residuals of U_x , implying that the results have converged.	34
4.7	Top: a numerical representation of the velocity distribution under the effect from a separation control, in the shape of a straight section, with $k - \omega$ modelling (Case VII). Bottom: a plot showing the residuals of U_x , implying that the results have converged.	35
4.8	Top: a numerical representation of the velocity distribution under the effect from a a straight section separation control mechanism with $k - \omega$ -SST modelling (Case VIII). Bottom: a plot showing the residuals of U_x , implying that the results have converged.	37
4.9	Top: a numerical representation of the velocity distribution under the effect of a straight section and with a Coanda jet implemented. The simulations was run with $k - \omega$ modelling (Case X). Bottom: plot showing the residuals of U_x , implying that the results have converged.	38

4.10	Top: a numerical representation of the velocity distribution under the effect of two separation control mechanisms, a straight section and a Coanda jet. The turbulence model chosen was $k - \omega$ -SST (Case XI). Bottom: plot showing the residuals of U_x , implying that the results have converged.	39
4.11	Case XIII showing the pressure field with swirl at the inlet. As the pressure recovery was the variable under investigation, the negative to zero scale was chosen to highlight this.	40
4.12	Case XIII showing the numerical representation of the velocity distribution for $k-\omega$ -SST profile and with swirl at the inlet. The inhomogenous field in the diffuser is further discussed in section 5.5. . . .	40
4.13	Time averaged pressure distribution in the domain for a simulation run using $k-\omega$ -SST modelling. The major concern for further investigation was the pressure recovery, which is the reason for the negative to zero scaling.	41
5.1	A comparison between three turbulence model and experimental data. Courtesy to Lo and Eaton [2] for the figure to the right of the experimental data. Used with permission	44
5.2	A comparison between two turbulence models and experimental data under the impact of passive separation control, implemented through an installed straight section. Courtesy to Lo and Eaton [2] for the figure to the right of the experimental data. Used with permission	46
5.3	A comparison between three turbulence models with experimental data under the impact of active separation control, implemented through a Coanda blow with a blow rate of 1. Courtesy to Lo and Eaton [2] for the figure to the right of the experimental data. Used with permission	48
5.4	Improved mesh	51
5.5	Old mesh	51
5.6	Close up picture on the jet/end region just behind the center body. Illustrates how the injected stream from the jet is unable to, according to the Coanda effect, follow the curvature of the end piece.	52
5.7	The attempt to better capture the Coanda effect by increasing the Coanda blow from 2.26 m/s to 5 m/s, making the blow ratio approximately 2.2	52
5.8	Heavily exaggerated Coanda blow rate (10 m/s giving a blow rate of approximately 4.4) manages to capture the Coanda effect along the tail piece. Additionally, the difficulty with using an O-grid to deal with the annular inlet is highlighted by the sudden jump in the separation region towards the end of the end piece.	53

List of Tables

3.1	Number of cells in mesh depending on geometry and turbulence model	22
4.1	Cases that has been simulated. The cases are combinations of setups where the presence of a straight section, a Coanda jet and swirl at the inlet has been varied.	27
A.1	y^+ values	I
A.2	Boundary conditions used	I

Nomenclature

- URANS: The Unsteady Reynold Averaged Navier Stokes equations.
- FVM: Finite volume method
- LES: Large Eddy Simulation
- SST: Shear Stress Transport
- SIMPLE: Semi-Implicit Model for Pressure-Linked Equations
- PISO: Pressure Implicit with Splitting of Operation

1

Introduction

The fundamental function of a diffuser is to, throughout the diffuser, recover the static pressure in exchange for a reduction of the dynamic pressure. A common case is when a center body exists before the diffuser. For example the exhaust diffuser behind a power turbine in a power generation plant. The bearing hub acts then as a center body that creates an annular inlet to the diffuser [2]. Similar setups are commonly encountered in various energy and propulsion systems. To highlight the need of an optimised diffuser design, roughly 47 % of the energy produced in Sweden came from hydro power in 2015 [1]. Therefore, as diffusers are present in a hydro power station, an optimised diffuser design could give a large effect on national matters.

The addition of a geometrical object such as a center body will make any attempt to analyse the flow, experimentally or numerically, more complex. The impact of the walls in such configurations furthermore complicates the analysis of the flow. The center body itself also impacts the flow in a way that causes separation, especially at high Reynolds numbers. Therefore, in terms of performance optimisation, implementation of separation control mechanisms are vital. Such control could be achieved via both passive and active methods, such as additions to the geometry of the diffuser and via the usage of Coanda jets respectively.

The ability to analyse methods to decrease separation is of interest. For many years, the standard way to do so was the experimental way. However, as computer performance has increased and as techniques and software to capitalise on that have been developed, the process of switching from experimental analysis to numerical simulations is under realisation. The benefits of moving into a reliable computerised way of working with the matter are vast as numerical simulations both are time-saving and cheaper compared to experimental work. The ability to test any new idea of how to, for instance, modify a diffuser to achieve separation control can be tested before a physical prototype is manufactured.

The computerised methodology is however under development, and a natural way to contribute to that development is through the comparison between the results obtained from a numerical simulation and existing experimental ones. In that way, by creating a digital mesh corresponding to the exact geometrical setup used by an experimental research team, various simulation techniques can be tested simultaneously. The variations in techniques are mainly focused around the choice of turbulence model and mesh quality to, ideally, fully capture the motion of the flow.

With this said, this project will aim to produce a comparison between the experimental results achieved by K. P. Lo and J. K. Eaton in their paper "Flow separation control for robust conical diffuser design" [2] and numerical results following simulations done by this team. The variations in the simulation will mainly be limited to different turbulence models, whereas the used software will be kept constant. The project will also involve the development of a reliable mesh, with resolution fine enough to give comparable results. To do so, both an understanding of the physics that lies in the governing equations for fluid motion in confined space with a centre body present and knowledge of the softwares (ANSYS ICEM CFD and OpenFOAM) will have to be developed.

1.1 Aim

The aim of this study is divided into three different areas; the physics of the fluid, the numerical simulation and a comparison of the results from the CFD-simulations to experimental data.

Firstly, the aim of this study is to understand the flow in a conical diffuser with an annular inlet and a centre body, with attached airfoils, present inside the diffuser. The aim is to understand the coupling between the wall boundary layer development and the flow separation. The separation in focus for this project is the so called "wake", a separation due to an adverse pressure gradient after the abrupt end of the centre body.

Secondly, the project aims to produce a three component time averaged velocity numerical simulation of the flow through the diffuser. This will be done using ANSYS ICEM CFD for mesh generation and OpenFOAM to solve Navier-Stokes equations using that mesh. Initially, the aim is to investigate two different geometries to see the difference in achieved separation control. OpenFOAM will then offer the chance to implement various turbulence models and inlet conditions. In terms of turbulence models, the following three will be the tested ones: $k - \omega$, $k - \omega$ -SST and $k - \omega$ -SST-SAS. Variations at the inlet will be limited to uniform flow and flow with swirl. The project also aims to implement and investigate the attempt to achieve separation control using Coanda jets at the end of the center body.

Thirdly, the results from the numerical simulations will be compared to the experimental database developed by Lo and Eaton [2].

In bullet form, the different cases that will be investigated are

- 2 different geometries (corresponding to cases II, III, IV and V in the report from Lo and Eaton)
- 2 different inlet velocity conditions (uniform and with swirl)

- The addition of an internal jet stream
- Each of the above mentioned cases with 3 different turbulence models ($k - \omega$, $k - \omega$ -SST and $k - \omega$ -SST SAS)

1.2 Limitations

As this project eventually aims to produce a comparison with experimental data, its setup will consequently try to mimic the cases in the report from Lo and Eaton. However, due to the fact that each simulation will be rather time demanding (an initial estimation is around 24 hours), we will limit our study to four of the 27 cases that Lo and Eaton looked into. The limitation is furthermore not purely due to computer power limitations, but also due to the fact that both ANSYS ICEM CFD and OpenFOAM are new programs for the investigating group, and will therefore initially require development of basic knowledge.

2

Theory

As stated in the aim section, one goal of this project is to develop a good knowledge of flow through the geometry at hand. Therefore, it is important to beside the numerical simulation look into the governing equations for flow in a diffuser. More specifically, as the eventual results will be based on numerical methods, the concept of discretisation (of the governing equations) is of particular interest. So is also the physics behind phenomenon such as separation and the Coanda effect, as both of those will be present in the simulations.

2.1 Navier Stokes equations to URANS

The understanding of the kinematic behaviour of the flow pattern in this case lies in solving Navier-Stokes equations for the specified geometry. Such analytical solution does of course not exist, whereof a numerical approximation has to be made. Numerically speaking, that means that Navier-Stokes equations

$$\rho\left(\frac{\partial \mathbf{u}}{\partial t} + (\mathbf{u} \cdot \nabla)\mathbf{u}\right) = -\nabla p + \mu \nabla^2 \mathbf{u} + \rho \mathbf{g}. \quad (2.1)$$

has to be rewritten on a discretised form.

Further on, the physics of the modelling lies in the solution to Navier-Stokes equations as in (2.1), and are here considered under the assumption of incompressible flow. Firstly, as a way to account for the unsteadiness and irregularity of turbulent flows, those equations are rewritten as the URANS, "The Unsteady Reynolds Averaged Navier-Stokes equations". URANS writes the velocity components as a sum of an averaged value (i.e. \bar{u}) and a fluctuation (i.e. u'). By introducing variables $u = \bar{u} + u'$, $v = \bar{v} + v'$, $w = \bar{w} + w'$ and $p = \bar{p} + p'$ where

$$\bar{u} = \frac{1}{t_2 - t_1} \int_{t_1}^{t_2} u d\tau \quad (2.2)$$

and using the fact that $\bar{\bar{a}} = a$ and $\bar{a'} = 0$ some modifications of (2.1) can be done.

Firstly, the time averaged equation of continuity gives

$$\frac{\partial(\bar{u} + u')}{\partial x} + \frac{\partial(\bar{v} + v')}{\partial y} + \frac{\partial(\bar{w} + w')}{\partial z} = \frac{\partial\bar{u}}{\partial x} + \frac{\partial\bar{v}}{\partial y} + \frac{\partial\bar{w}}{\partial z} + \frac{\partial u'}{\partial x} + \frac{\partial v'}{\partial y} + \frac{\partial w'}{\partial z} \quad (2.3)$$

which can be time averaged to obtain

$$\frac{\partial\bar{u}}{\partial x} + \frac{\partial\bar{v}}{\partial y} + \frac{\partial\bar{w}}{\partial z} = 0. \quad (2.4)$$

This result is then used when modifying (2.1) (without gravitational impact as the pipe where the simulation will be done is of sufficiently small diameter for that assumption to be made). For the direction of \hat{x} , the following is achieved

$$\begin{aligned} \rho \left(\frac{\partial(\bar{u} + u')}{\partial t} + (\bar{u} + u') \frac{\partial(\bar{u} + u')}{\partial x} + (\bar{v} + v') \frac{\partial(\bar{u} + u')}{\partial y} + (\bar{w} + w') \frac{\partial(\bar{u} + u')}{\partial z} \right) = \\ = -\frac{\partial(\bar{p} + p')}{\partial x} + \mu \left(\frac{\partial^2(\bar{u} + u')}{\partial x^2} + \frac{\partial^2(\bar{u} + u')}{\partial y^2} + \frac{\partial^2(\bar{u} + u')}{\partial z^2} \right). \end{aligned} \quad (2.5)$$

By expanding all the derivatives, using the two rules regarding time average of \bar{u} and u' , the results from equation (2.3) and some algebra, one eventually ends up with

$$\rho \left(\frac{\partial\bar{u}}{\partial t} + \bar{u} \frac{\partial\bar{u}}{\partial x} + \bar{v} \frac{\partial\bar{u}}{\partial y} + \bar{w} \frac{\partial\bar{u}}{\partial z} \right) = -\frac{\partial\bar{p}}{\partial x} + \mu \left(\frac{\partial^2\bar{u}}{\partial x^2} + \frac{\partial^2\bar{u}}{\partial y^2} + \frac{\partial^2\bar{u}}{\partial z^2} \right) - \rho \left(\frac{\partial(\overline{u'u'})}{\partial x} + \frac{\partial(\overline{u'v'})}{\partial y} + \frac{\partial(\overline{u'w'})}{\partial z} \right) \quad (2.6)$$

which is the RANS-form for the \hat{x} -direction. Note the last term, the Reynolds stress, which is an important addition to equation (2.1) that arises from this derivation. The equations for the y and z directions are achieved in an analogue fashion.

Eventually, we arrive at the URANS equations by noting that any unsteady behaviour will be captured through the transient term $\frac{\partial u_i}{\partial t}$.

Therefore, by combining the results of (2.6) (for all coordination directions) and adding the continuity equation, the complete set of the URANS equation on tensor notation are listed in equations (2.7) and (2.8).

$$\frac{\partial\bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j} (\bar{u}_i \bar{u}_j) = -\frac{1}{\rho} \frac{\partial\bar{p}}{\partial x_i} + \nu \frac{\partial^2\bar{u}_i}{\partial x_j \partial x_j} - \frac{\partial\overline{u'_i u'_j}}{\partial x_j} \quad (2.7)$$

$$\frac{\partial\bar{u}_i}{\partial x_i} = 0 \quad (2.8)$$

2.2 Turbulence modelling

Arriving at equation (2.7), one realises that these equations are not closed ones (i.e. the number of unknowns exceeds the number of equations, commonly known as the

"RANS closure problem"). Consequently, there is a need to model the turbulence in order to achieve a closed set of equations. More specifically, the task at hand is to model the Reynolds stresses, since that is where the turbulent impact lies. One of the methods to do so is through the Boussinesq approximation. The Boussinesq approximation postulates that the Reynolds stresses τ_{ij} (on tensor notation) relates to the mean strain rate tensor S_{ij} and the turbulent kinetic energy k in the following way:

$$\tau_{ij} = 2\nu_t S_{ij} - \frac{2}{3}\rho k \delta_{ij} \quad (2.9)$$

where ν_t represents the turbulent eddy viscosity and ρ the density. It is worth underlining the fact that ν_t , in contrast to its molecular equivalence, not can be seen as generally constant throughout the domain. It should be noted that ν_t is a property of the flow and not the fluid.

Like any model, the Boussinesq approximation has advantages and disadvantages. The advantages lies in its simplicity, where such simplification especially appeals to the opportunity to create the analogy between turbulence and molecular viscosity. The disadvantage is, like in many approximations, the range of situations in which it is valid. This one especially collapses as the flow accelerates and decelerates as well as for flows in strong curvatures.

Furthermore, in this project, the Boussinesq approximation will be implemented through the two-equation turbulence models. According to Argyropoulos and Markatos [11], the notation "two-equation" refers to how these models calculates the turbulent viscosity ν_t . Dimensional analysis indicates that $\nu_t \propto V_t L_t$, where the latter is a product of a characteristic velocity and a characteristic length scale. In the zero equation model, both V_t and L_t is calculated through algebraic expressions, whereas the one-equation and two-equation models calculates V_t and L_t as well as V_t respectively via differentials.

Specifically, two-equation models utilises two additional transportation equations to close the problem. Examples of transportation equations are for turbulent kinetic energy equation k , turbulent dissipation ϵ and specific dissipation ω , which combines to especially the k - ϵ and k - ω models.

2.2.1 RAS turbulence models

To limit this theory section, only models which further on will be implemented or subjects for discussion will be mentioned here. The reason the two-equation model is used is, amongst other, the fact that it arguably is the most used one in industry and in academia, and consequently providing a good amount of literature. A noteworthy advantage with working with k - ω instead of k - ϵ models is that k - ω behaves better in areas with a weak adverse pressure gradient [6], a phenomenon that according to experimental data is to be expected in the wake behind the center body.

2.2.1.1 k - ω model

k - ω is the first turbulence model used in this simulation that will be discussed. The model is among the most commonly used two-equation version and closes the Navier-Stokes equations in (2.7) by modelling of the turbulent kinetic energy k and the specific dissipation rate ω .

$$(\rho \bar{u}_j k)_j = \left[\left(\mu + \frac{\mu_t}{\sigma_k} \right) k_j \right]_j + P_k - \beta^* \omega k \quad (2.10)$$

$$(\rho \bar{u}_j \omega)_j = \left[\left(\mu + \frac{\mu_t}{\sigma_\omega} \right) \omega_j \right]_j + \frac{\omega}{k} (c_{\omega 1} P_k - c_{\omega 2} \rho k \omega) \quad (2.11)$$

$$\mu_t = \rho \frac{k}{\omega}, \varepsilon = \beta^* \omega k. \quad (2.12)$$

$$\varepsilon = \frac{k^{\frac{3}{2}}}{l} \quad (2.13)$$

$$P_k = \rho \frac{u_*^{3*}}{\kappa y} \quad (2.14)$$

which together with $P_k = \rho \varepsilon$ gives:

$$l = \kappa c_\mu^{\frac{-3}{4}} y \quad (2.15)$$

The constants can from the above equations be determined as following:

$$\beta^* = 0.09, c_{\omega 1} = 5/9, c_{\omega 2} = 3/40, \sigma_k^\omega = 2 \text{ and } \sigma_\omega = 2$$

The case when wall functions are used, k and ω are chosen as:

$$k_{wall} = (\beta^*)^{\frac{-1}{2}} u_*^2, \omega_{wall} = (\beta^*)^{\frac{-1}{2}} \frac{u_*}{\kappa y} \quad (2.16)$$

In special cases when turbulence is low in some regions, that is when both k and ω approaches zero, it introduces large numerical problems for the k - ε model as k becomes zero. This problem does not appear in the ω equation, which makes this model beneficial compared to the k - ε model. If k approaches zero, the turbulent diffusion term simply goes to zero [4]. To be mentioned is that the production term in the ω equation does not include k , this is shown in equation 2.17.

$$\frac{\omega}{k} c_{\omega 1} P_k = \frac{\omega}{k} c_{\omega 1} \mu_t \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \frac{\partial \bar{U}_i}{\partial x_j} = c_{\omega 1} \beta^* \left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) \frac{\partial \bar{u}_i}{\partial x_j} \quad (2.17)$$

2.2.1.2 $k - \omega$ -SST model

The second turbulence model the simulation will use is the SST (shear stress transportation) $k-\omega$ model. The model was first proposed by F.R. Menter in 1994. Mathematically speaking, the model is implemented through two transportation equations, accounting for the turbulent kinetic energy (2.18) and the rate of dissipation of the eddies (2.19) respectively. Those are

$$\frac{\partial k}{\partial t} + u_j \frac{\partial k}{\partial x_j} = P_k - \frac{9}{100} k\omega + \frac{\partial}{\partial x_j} \left[(\nu + \sigma_k \nu_T) \frac{\partial k}{\partial x_j} \right] \quad (2.18)$$

$$\frac{\partial \omega}{\partial t} + u_j \frac{\partial \omega}{\partial x_j} = \alpha S^2 - 0.075\omega^2 + \frac{\partial}{\partial x_j} \left[(\nu + \sigma_\omega \nu_T) \frac{\partial \omega}{\partial x_j} \right] + 2(1 - F_1) \sigma_{\omega^2} \frac{1}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i} \quad (2.19)$$

with

- $P_k = \min\left(\tau_{ij} \frac{\partial u_i}{\partial x_j}, \frac{9}{10} k\omega\right)$
- $\nu_T = \frac{\frac{5}{9} k}{\max(\frac{5}{9} \omega, SF_2)}$
- $F_1 = \tanh\left(\left(\min\left(\max\left(\frac{\sqrt{k}}{\frac{9}{100}\omega y}, \frac{500\nu}{y^2\omega}\right), \frac{3.424k}{CD_{k\omega}y^2}\right)\right)^4\right)$
- $CD_{k\omega} = \max\left(1.712\rho \frac{1}{\omega} \frac{\partial k}{\partial x_i} \frac{\partial \omega}{\partial x_i}, 10^{-10}\right)$
- $F_2 = \tanh\left(\left(\max\left(\frac{2\sqrt{k}}{\frac{9}{100}\omega y}, \frac{500\nu}{y^2\omega}\right)\right)^2\right)$

The $k-\omega$ -SST model could be seen as a combination between $k-\omega$ and $k-\epsilon$. $k-\omega$ is known to be better than $k-\epsilon$ in predicting the flow in separated regions. However, as $k-\omega$ is dependent on the free stream value of ω , one could see the advantages in developing a combined model for the two. The combination lies in a switch between the two models to active each one in the region they perform well respectively. Hence, $k-\omega$ -SST models the flow with $k-\omega$ behaviour at small distances from the wall (through the vicious sub-layer). As the distance from the wall grows, it switches to $k-\epsilon$ to avoid its dependency on the free stream value of ω .

The $k-\omega$ model is especially bad in adverse pressure regions, a situation that is highly likely to be encountered in the geometry for this project. The adverse pressure region will be developed in the inner boundary layer, so in order to improve the $k-\omega$ -model, $k-\omega$ -SST will apply a $k-\epsilon$ -behaviour in that region. However, it is favourable to keep the behaviour of the $k-\omega$ -model at the outer parts, since a weakness with $k-\omega$ is that it depends on the free stream value of ω [6]. A second advantage of the *SST* version is that it, in contrast to the $k-\omega$ -model, avoids overestimating the shear stress (thereof the name) in regions with adverse pressure gradient. The result of those two improvements in the mathematical sense are the equations presented in

(2.18) and (2.19), whereas the difference in simulation quality is to be determined and hopefully reviewed in the result section.

2.2.1.3 k - ω -SST-SAS

The third and final turbulence model this project will cover is the k - ω -SST-SAS model. This model is one of various SAS (scale adaptive simulation) techniques, but was chosen naturally as a further extension of the previous models ($k - \omega$ and $k - \omega$ -SST). In short, the SAS features are added to the $k - \omega$ -SST model to acquire a more realistic model of the flow by creating a model that behaves in different ways at different distances from a wall. The reason this is achieved is because of several factors. Firstly, the fact that if a flow shows a tendency to go unsteady, the chosen model should rather be able to capture that than force the flow to remain steady. For instance, if the solver only looks for steady solutions to the problem, situations with unsteady flows will not be correctly modelled as a solution do not exist. Secondly, conventional URANS models (like $k - \omega$ -SST) will dampen out such unsteady behaviour if the turbulent viscosity is sufficiently high. Therefore, a desirable feature for the SAS-model is its ability to resolve large-scale motions [6], instead of modelling them.

In order to not become unsteady, a production term P_ω is added to the ω equation (2.19) of the $k - \omega$ -SST model that will provide an increase in ω when unsteadiness appears. Unsteadiness is associated with a local increase in velocity gradients, which furthermore decreases the turbulent length scale. The chain of events then is that a decreased turbulent length scale will result in a decreased von Karman length scale $L_{\nu K, 3D}$. Hence, if $P_\omega \propto 1/L_{\nu K, 3D}$ and $\nu_t \propto 1/\omega$, P_ω will increase as $L_{\nu K, 3D}$ decreases causing ω to increase and eventually ν_t to decrease. For more information see [15].

An effect of the behaviour above is, according to a group of scientists at Florianopolis University [14], that SAS behaves like a RANS model in boundary layer regions, and a LES model in detached regions. Consequently, one can make the analogy to the $k - \omega$ -SST-model in that it changes its characteristics as the distance from walls increases.

2.3 Discretisation

The discretisation can be thought of to involve two main parts. The first one discretises the geometry into a set of interacting volumes, also known as the mesh. The second one is how the mathematical operations and governing equations can be discretised in order to enable computational power to aid in solving the set of resulting equations.

The finite volume method (FVM for short) is fundamentally based on the concept

of control volume integration to account for the flux of the relevant quantity.

2.3.1 Geometric discretisation

Initially, the discretisation process is a matter of mesh generation. The resulting mesh creates a grid of cells interacting throughout the physical domain.

Firstly, the scientific way to divide the domain into sub-domains is based on the following nomenclature. For a three-dimensional case, the cells surrounding a cell with centre point P are denoted W (west), E (east), N (north), S (south), T (top) and B (bottom) where the positive axes (in a conventional Cartesian coordinate system) are in the E, N and T direction. The exception is when a cell is located adjacent to a boundary. Then, (at least) one face is under other physical constraints set by the type of boundary and model chosen.

Secondly, interactions between cells is a feature the mesh generation has to account for. The most important fundamental physical law the mesh generation relies on is the conservation of transport properties. Therefore, the interactions between cells are highly important and occurs over the faces. As the physical domain the grid is created in is arbitrary in the general case, the mesh will not be uniform shape-wise. Concepts such as aspect ratio and orthogonality are measurements on how well the mesh will handle the conservation between cells. Aspect ratio could be taken both as the size ratio of adjacent edges in the same direction (according to chosen coordinate system) or the ratio between the edges of the same cell. Orthogonality is usually measured as the angle between edges in one cell. Ideally, that angle is always 90 degrees, which limits the possible loss of any component in the interaction between cells.

Eventually, the connection between cells forms local matrices which then are put into a global matrix. The global matrix provides consistency across all cells and is rather based on the indices of the cells than the cardinal direction nomenclature used for local assembly.

2.3.2 Equation discretisation

Discretisation of the partial differential equations (PDEs) that describes the motion of the fluid is essential in order to be able to use computational power to aid the solving. A general discretisation could be said to be an operation that takes a PDE and transforms it into a set of linear equations $Ax = b$. Such system of equations is then solvable using, for instance, OpenFOAM.

In OpenFOAM, the discretisation method is specified when choosing a appropriate numerical scheme. The numerical schemes are protocols specifying how interpolations, gradients, laplacians, divergence and time dependence are approximated. For

example, Gauss finite volume integration is a common scheme for discretisation of derivative terms. This scheme is based on the Gauss theorem, in which numerical values are given by summing all values on the cell faces. Such summation, however, has to interpolate the values on those faces from the cell centre. Examples of interpolation methods are centred and upwind schemes (based on central differentiating and upwind differentiating respectively).

There are various schemes for each of the mathematical operations that has to be discretised and how they are realised by numerical means, but it is beyond the scope of this thesis to treat those. For more information about discretisation of equations relevant to fluid mechanics see [3].

2.3.3 Solver selection

It is beyond the scope of this thesis to thoroughly explain the inner workings of the solver used in this project. Instead, a short description for the choice of solver for this project will follow in the next subsection.

2.3.3.1 PIMPLE

The used algorithm in this project is the PIMPLE algorithm. PIMPLE is a merged SIMPLE and PISO solver and has a few advantages over each of them. Especially, the `nNonOrthogonalCorrectors` feature accounts for the loss across faces of cells that are not orthogonal. This is to be expected in this project since the geometry is quite complex. Furthermore, PIMPLE is able to take big time steps in order to reduce the computing demand but simultaneously is allowed to do so without any complaints about the PISO algorithm's aim to achieve stability by having a Courant number $Co < 1$ [23] using relaxation factors (a SIMPLE feature) between time steps.

2.3.4 Courant condition

One important parameter in terms of stability for several solvers is the Courant condition. As noted in previous section; PISO aims to fulfil $Co < 1$, where the condition for incompressible flow can be defined as

$$\frac{u\Delta t}{\Delta x} < \alpha. \quad (2.20)$$

Here α is a parameter that depends on the particular time-advancement method used, Δt denotes time step, u is the magnitude of the velocity and Δx is the length interval [23].

2.4 Characteristics in boundary layer

This section will give a description of how the boundary layer, fluid flow under the influence of walls, is structured. The reason for including this section is primarily that when turbulence is modelled, it is of high importance to know both the physical characteristics of the boundary layer and how different turbulence models treat the boundary layer as this differs widely (see section 3.1.2 for more information about this). In fluid dynamics it is often preferred to use dimensionless quantities in order to make things more generalised. For boundary layer flow it is common to use the dimensionless quantity y^+ . This quantity is mainly used to show how the flow characteristics change depending on the distance from the wall. The equation for y^+ reads

$$y^+ = \frac{yu_\tau}{\nu} \quad (2.21)$$

where u_τ denotes the friction velocity.

The region where the boundary layer exists can be divided into an inner region and an outer region as seen in figure 2.1.

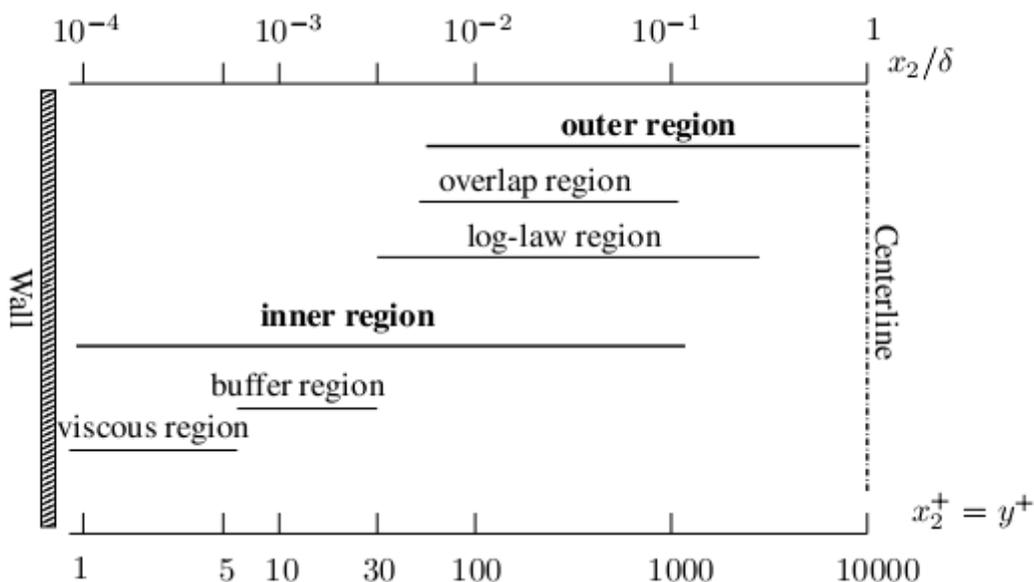


Figure 2.1: The buildup of different regions of the boundary layer near a wall. (Credits for the image goes to L. Davidson [6]. Used with permission.)

In the viscous region, the flow is dominated by viscous shear stresses caused by flow interactions with the wall. Further out from the wall has the turbulent stresses increasingly greater impact and the wall shear stress less impact. This transition region is called the buffer region; after which, the logarithmic region follows where

turbulent stress dominates. Eventually, the so called outer region starts. For a graph over how the flow behaves near a wall see figure 2.2.

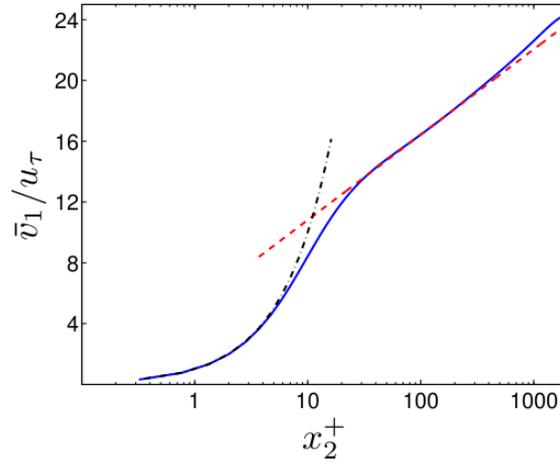


Figure 2.2: Graph over the flow behaviour near the wall where the regions for where the viscous layer model (the dashed black line) and the logarithmic law model (the dashed red line) are valid can easily be seen. The blue line is data from a DNS (Direct Numerical Simulation). (Credits for the image goes to L. Davidson [6]. Used with permission.)

In the viscous layer is $\bar{v}_1^+ = y^+$, where \bar{v}_1^+ is a dimensionless "velocity" scaled with quantities related to the wall. Another way to write (2.4) in the viscous layer is (2.22).

$$\bar{v}_1^+ = \frac{\bar{v}_1}{u_\tau} = \frac{u_\tau y}{\nu} \quad (2.22)$$

Further away from the wall, the fluid follows the model of the log-law closely where now \bar{v}_1^+ can be written as in equation (2.23).

$$\bar{v}_1^+ = \frac{1}{\kappa} \ln\left(\frac{u_\tau y}{\nu}\right) + B \quad (2.23)$$

where B is an integration constant. The constants in the log-law are usually set to $\kappa = 0.41$ and $B = 5.2$. For full derivation of the log law see Lars Davidson's work on the matter [6].

2.5 The separation phenomenon

Flow in a confined space, such as the pipe considered in this project, is subject to possible separation. Especially, this can be the case when a body of some sort (here a centre body) abruptly ends. As stated in the introduction, separation can

create losses in diffuser performance, and is therefore an important part of the theory describing fluid in motion in the given geometry. A possible way to work with active separation control is through Coanda blow, a method that has been implemented and will be described further in section 3.1.3.1.

2.5.1 Separation

Initially, the authors once again point out the importance of including theory of actual (direct) relevance for this project. With that in mind, the phenomenon known as separation is of particular interest as the geometry at hand is favourable for separation development.

Separation is, fundamentally speaking, a result of an adverse pressure gradient ($\frac{dp}{dx} > 0$) in the boundary layer. The theoretical connection between the pressure gradient and development of separation is to be found in the derivation of the “boundary layer equations”. The conversion of momentum follows in equation (2.24).

$$u \frac{\partial u}{\partial x} + v \frac{\partial u}{\partial y} \approx U \frac{dU}{dx} + \frac{1}{\rho} \frac{\partial \tau}{\partial y}. \quad (2.24)$$

Close to the wall, $u \approx v \approx 0$, which via potential theory and Bernoulli’s equation gives

$$\left. \frac{\partial \tau}{\partial y} \right|_{\text{wall}} = -\rho U \frac{dU}{dx} = \frac{dp}{dx}. \quad (2.25)$$

Further on, close to the wall, all flow is laminar, i.e. τ follows the laws as in equation (2.26).

$$\begin{aligned} \tau = \mu \frac{\partial u}{\partial y} &\Rightarrow \left. \frac{\partial \tau}{\partial y} \right|_{\text{wall}} = \mu \left. \frac{\partial^2 u}{\partial y^2} \right|_{\text{wall}} \\ &\Rightarrow \left. \frac{\partial^2 u}{\partial y^2} \right|_{\text{wall}} = \frac{1}{\mu} \frac{dp}{dx}. \end{aligned} \quad (2.26)$$

In other words, if there is an adverse pressure gradient, the second derivative of the velocity (in the x-direction in conventional nomenclature) is also greater than zero. But, if $u \rightarrow U$ in the outer parts of the boundary layer, the second derivative must be negative. The only possible solution to those two restrictions is the existence of an inflexion point somewhere along increasing values of y . Such S-shaped velocity profile could then result in a separation close to to the wall.

For the case presented in this thesis, Eaton [2] predicts there to be a coupling between the formation of the separation bubble after the centre body and the formation of the wall boundary layer. If the separation bubble closes as an effect of separation control methods, the flow is predicted to be suspect to an adverse pressure gradient and susceptible to flow separation. For the case when the separation bubble after the centre body exists, the flow is less susceptible to flow separation near the wall because of blockage effect from the bubble which relieves the adverse

pressure gradient. For optimal diffuser performance though, the separation bubble should still be minimised in size or closed since the bubble has its own negative effects on the pressure recovery and flow uniformity.

2.5.2 Coanda effect

The Coanda effect (named after Henry Coanda, a Romanian inventor) is the tendency for a fluid to attach and follow a curved surface [7]. The effect was noticed by Henry Coanda in 1910 when he noticed that when air was ejected from a rectangular nozzle, it would attach itself to an inclined flat plate connected to the nozzle exit. This effect has been found to occur for both liquids and gases but is only correctly called the Coanda effect if the jet and the ambient fluid is of the same phase [8] (this means for example that the phenomena which occurs when holding a spoon under the water jet in a kitchen sink is *not* the Coanda effect since the ambient fluid, air, is in gas-phase).

When a jet of a fluid moves through ambient fluid the jet entrains the molecules near the jet, causing a low-pressure region to form around the jet. This causes the surrounding ambient air of higher pressure to exert a force on the low pressure region close to the jet. If a plate is placed horizontally close to the jet there will be a region of low pressure close to the wall since the air close to the surface can not neutralise its pressure with ambient air. This will cause the jet to adhere to the surface. If the plate has a curved surface the jet will follow the curved surface unless the curve gradient is too steep and separation occurs.

This effect offers an important tool to achieve separation control, as the tendency to follow the curved end surface of the centre body possibly could counteract the separation wake. The concept of implementing this theory is further described in the method section.

3

Method

The method section aims to describe the working process used in general, and more specifically point out the important choices and methods used to eventually produce the results. The process in itself was divided into three main categories. The first one is centred around the generation of the a mesh using the software ANSYS ICEM CFD. The second one describes how to do the numerical calculations using the CFD solver OpenFOAM, with the visual representation aid from ParaView. The third presents the post-simulation tools used to produce relevant visualisations of the simulation, aimed to be compared with the experimental results from Lo and Eaton.

3.1 Generation of mesh

First of all, it is worth mentioning that the generation of the mesh truly is a matter of quality. The aim of this paper is to numerically simulate the process of the flow through a given geometrical setup, and furthermore attempt to compare achieved results with experimental data. Therefore, the dilemma at hand lies between the fact that a large number of cells in the mesh increases the quality, but will on the other hand increase simulation time. Additionally, one of the turbulence models (SAS) used requires an increased mesh quality compared to the other two, meaning that two meshes for each of the geometric setups had to be produced.

3.1.1 Geometrical setups

To begin with, a careful evaluation of the geometric setup used by Lo and Eaton was done. Their paper [2] presents a very detailed description of the diffuser, which gave sufficient information for a computerised version to be produced. Lo and Eaton did experiments on a vast amount of geometrical setups (and with different inlet conditions and separation control features). Therefore, as each simulation will take in the order of tens of hours, the study was narrowed down to (with Lo and Eaton's nomenclature) case II, III, IV and V [2]. Those cases are further on explained below.

Firstly, speaking in terms of geometric variations, those four cases are based on two different setups.

3.1.1.1 Geometric setup without straight section

To begin with, this setup was the one used in the study of the cases with geometries without a straight section (in contrast to the second geometry which used a straight section as a passive separation control). This geometry is called geometry 1. The geometry of the problem at hand could be described as a diffuser with an annular inlet and center body present. Consequently, it is natural to separate the description of the geometry into two different, geometrically independent from each other, parts: the diffuser and the center body.

The diffuser contains of the six parts listed below. The contraction and diffuser part of those are illustrated in figure 3.1.

- A 560 mm long annular inlet section with a diameter of $D_{inlet} = 76$ mm.
- A 115 mm long contraction reducing the diameter from D_{inlet} to $D_{center} = 60.7$ mm.
- A 65 mm long straight section with a diameter $= D_{center}$.
- The first (of two) diffuser sections is 30 mm long and increases the diameter from D_{center} to $D_{diffuser,1} = 66.5$ mm.
- The second diffuser section is 106 mm long and increases the diameter from $D_{diffuser,1}$ to $D_{diffuser,2} = 94.6$ mm.
- A 420 mm long outlet with a diameter $D_{diffuser,2}$.

Whereas the diffuser is a simple geometric setup with only straight lines and a varying diameter, the center body used was more complex. The entire body was 131.4 mm long and could be said to be composed by three main parts (in this order from up- to downstream)

- A nose in the shape of an ellipse with a minor/major radius composition of 20 and 50 mm respectively.
- A 60 mm long cylindrical section with a diameter of 33.3 mm.
- A 26.4 mm long Coanda tailpiece with a minor/major radius composition of 15.4 and 16.4 mm respectively.

A cut of the pipe across the section where the center body is is presented in figure 3.1, and a figure of the same center body only is to be seen in figure 3.3.

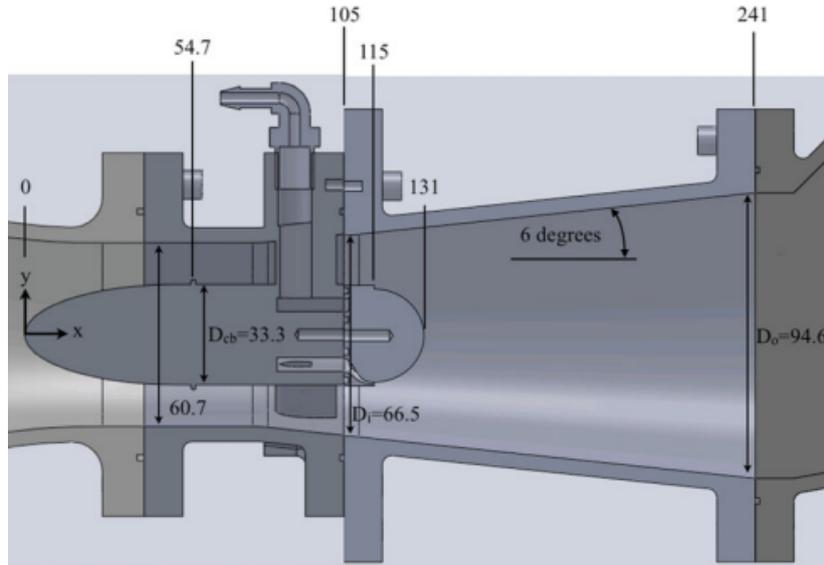


Figure 3.1: A detailed description of the measurements used for the center body and the surrounding outer wall section. Amongst other, a noteworthy feature is the fact that the coordinate system used in this work as well as in the experimental project has its origin at the nose of the center body. Credits for the image goes to Eaton and Lo [2]. Used with permission.

Furthermore, there are some extra geometrical features to the center body. Firstly, an O-ring was added circumferentially at the end of the nose (50 mm downstream of the start of the center body). The O-ring adds an additional 1.6 mm to the diameter of the cylindrical part, making it 34.9 mm. It is worth noting that the usage of O-rings aims to induce a turbulent flow along the center body.

Secondly, as the cylindrical part ends and transitions into the Coanda tail piece, the ability to eventually implement Coanda blow was realised. The diameter of the cylindrical part was 33.3 mm and the minor radius of the Coanda tailpiece is 15.4 mm. This leaves a $33.3 - 2(15.4) = 2.5$ mm vertical (in the natural coordinate system having the x along the stream) gap. This gap was filled with a 0.55mm thickness of the cylindrical part and, more relevant for the study, a 0.7 mm thick Coanda blow section, as visualised in figure 3.2. The Coanda blow could therefore be realised with a boundary condition in OpenFOAM corresponding to the blow rate wanted.

The blow rate was taken from the report of Lo and Eaton to match the one used by them. They defined the blow rate (BR) as

$$BR = \frac{U_{jet,bulk}}{U_{bulk}} \quad (3.1)$$

meaning that given a $BR = 1$ (the one used), the boundary condition implemented at the jet patch of the mesh was simply U_{bulk} , which was given in the report to be 2.26 m/s. The bulk velocity was taken at the annulus section upstream of the five

center body struts [2].

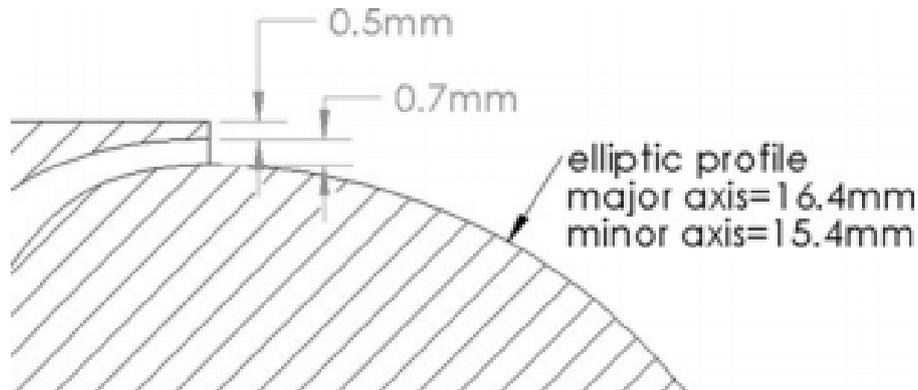


Figure 3.2: A figure to illustrate how the Coanda jets were implemented in the experimental setup. This setup was mimicked in a numerical sense by the creation of a separate geometric pieces representing the jets, where furthermore the effect of the jets could be implemented through a boundary condition corresponding to the blow rate wanted. Credits for the image goes to Eaton and Lo [2]. Used with permission.

Thirdly, the center body was held in place by five hollow NACA 0024 airfoils, functioning as struts to support the center body. The airfoils were 8 mm long and start 82.5 mm downstream of the nose. The strut geometry was chosen to mimic typical shapes and blockages of the struts used in large-scale power generation turbines [2].

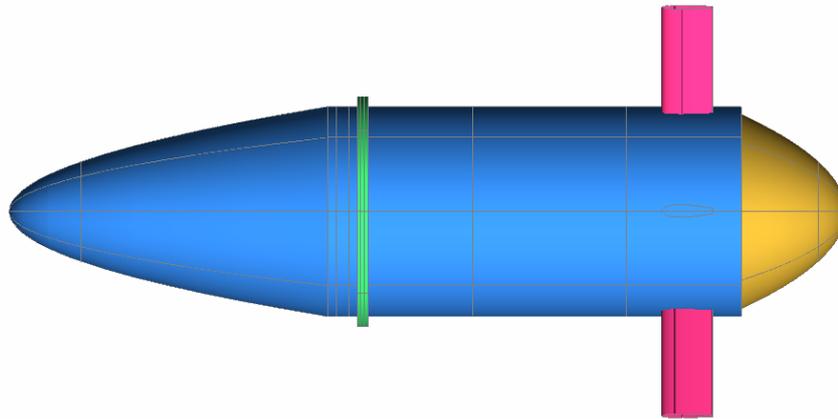


Figure 3.3: A visualisation of the center body used in the experiment. Relevant parts to notice are the O-ring (green) and the Coanda tailpiece (yellow).

3.1.1.2 Geometric setup with straight section

This geometry is called geometry 2. Unlike geometric setup 1, this setup had a 2 center body diameter long straight-wall section installed between the center body and the second diffuser part as illustrated in figure 3.5. The generated geometry

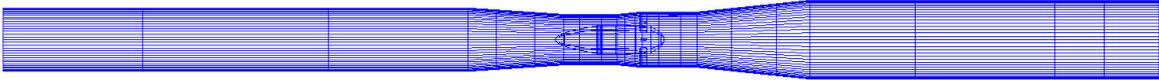


Figure 3.4: A view of the second geometry generated in ICEM CFD, with an elongated cylindrical section, seen from the side. The measurements can be found listed in the text.

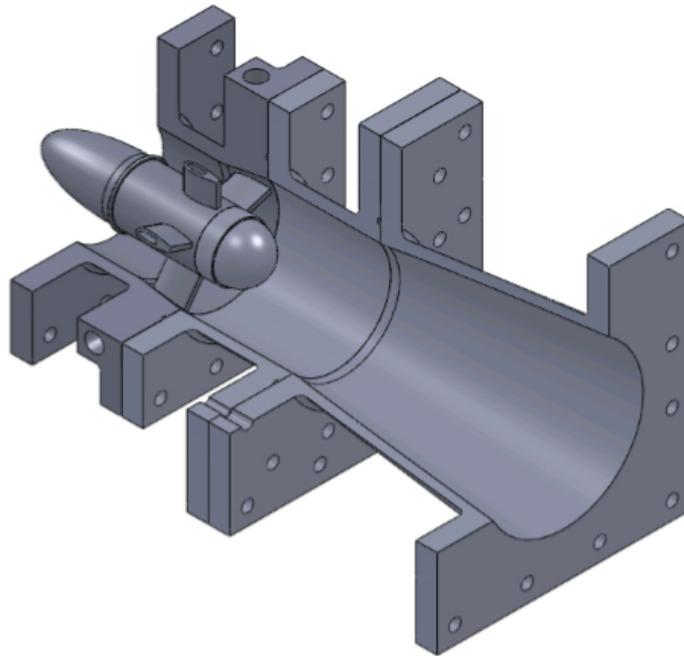


Figure 3.5: A schematic view of the geometry with the straight section implemented after the center body. Credits for the image goes to Eaton and Lo [2]. Used with permission.

is shown in 3.4. In theory, and confirmed by the report from Lo and Eaton, such straight-wall section allows the flow to smoothly transition without any pressure gradient discontinuity and consequently allow the separation bubble to, locally, close naturally. As for negative effects, this can result in a bigger and more expensive structure. This is a form of passive flow control since the addition is always active and is not possible to control.

3.1.2 Mesh quality and y^+

As previously stated, the mesh quality is a delicate matter of receiving comparable results within the time frame for the project. Furthermore, the quality of the mesh can be related to two features.

Firstly, mesh quality will inevitably rely on the number of cells in the mesh. More cells will result in a finer resolution and consequently give a better possibility to achieve results that better represent the natural continuous motion of fluids. However, if the number of cells globally were to be unchanged, mesh generation is still a matter of concentrating the cells to the regions relevant for the eventual investigation. In this project, the separation in the wake of the center body is the main area of interest. Therefore, a majority of the cells will be located there. The total number of cells used for the two cases and for the two geometries are presented in table 3.1. Note that the mesh for $k-\omega$ -SST-SAS for the second geometry is smaller than the one used for SAS simulations on geometry 1. The reason for this is that a lot of extra work was put into the first case to make that work before moving onto geometry 2. As no successful simulations were obtained for the SAS simulations in the first case, such improvement of the SAS mesh for geometry 2 was never made. Hence, the smaller mesh in terms of number of cells.

Table 3.1: Number of cells in mesh depending on geometry and turbulence model

Geometry	Turbulence model	Nbr. of cells
1	$k-\omega/k-\omega$ -SST	233860
1	$k-\omega$ -SST-SAS	1759008
2	$k-\omega/k-\omega$ -SST	609460
2	$k-\omega$ -SST-SAS	1182288

Secondly, the mesh generation has to take into account what turbulence model later on will be implemented. As stated above, this project used $k-\omega$, $k-\omega$ -SST and $k-\omega$ -SST-SAS to model the turbulence. In terms of mesh structure, they are quite different when it comes to spacing around the walls. The first two, $k-\omega$ and $k-\omega$ -SST, are models that use wall functions that are given by OpenFOAM. For instance, the wall function for the kinematic turbulent energy k is implemented in OpenFOAM via the *kqrWallFunction* option. Therefore, the mesh for a simulation with $k-\omega$ (with or without -SST) has to have a $y^+ > 30$ in order to not inflict with the inner region (see theory section on y^+). For $k-\omega$ -SST-SAS, however, the turbulence model (as part of the LES family of models) resolves the viscous part of the inner layer which means that the mesh has to be so fine that the first cell does not exceed that layer, i.e. $y^+ < 4$ and ideally $y^+ \approx 1$. Logically, this will result in a finer mesh and consequently increase the number of cells globally.

To illustrate this, the difference between the number of cells as well as the spacing around the walls is visualised in figure 3.6 and figure 3.7.

3.1.3 Implementation of flow control

In the following section, in addition to the passive flow control that consists of the straight section geometry addition discussed in section 3.1.1.2, one active type of flow control and one additional passive type of flow control will be described.

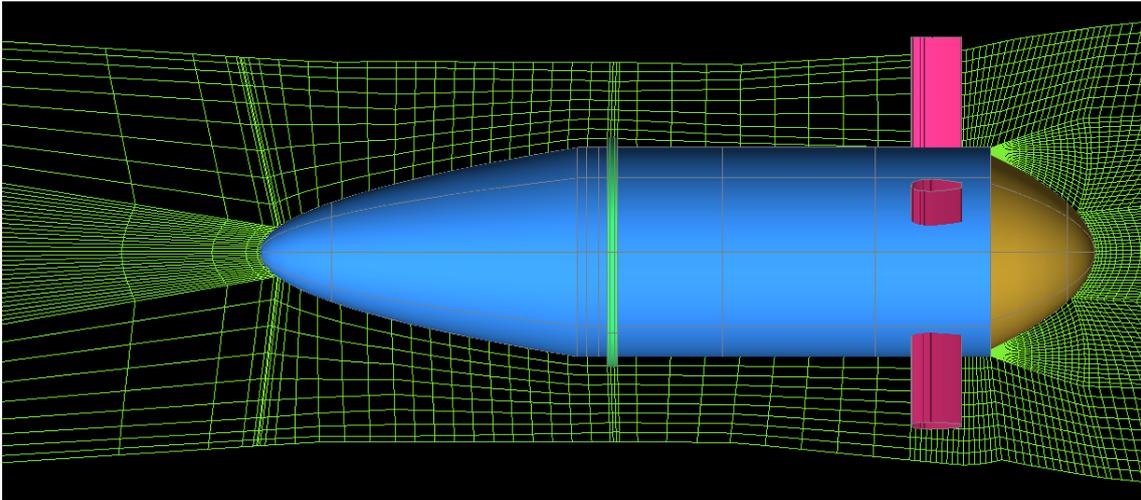


Figure 3.6: A visualisation of the mesh used for $k-\omega$ and $k-\omega$ -SST simulation. The mesh contains approximately 250000 cells and a y^+ at least greater than 20 globally and locally fulfills $30 < y^+ < 100$ locally. The shot is taken with the "plane function", which failed to give a complete view of the cells close to the center body wall.

3.1.3.1 Coanda blow as an active flow control

Active flow control is characterised by the fact that it can be regulated and that it consumes energy when used [22]. The type of active flow control utilised in this project is Coanda blowing.

The theory behind the Coanda effect was presented in section 2.5.2. The reason a jet is installed at the end of the center body is to achieve separation control via the physical concept of the Coanda effect. The Coanda effect states that when a gas or liquid passes over a curved surface, it will follow the surface due to the development of low pressure regions [7]. In other words, a Coanda jet located at the end of the center body will inject a water stream to the flow that will follow the Coanda tailpiece and consequently counteract the separation process. This is visualised in figure 3.8.

As stated, a numerical realisation of the Coanda jets will be to add a geometrical object (here a circumferential ring around the end piece) where a boundary condition can be implemented. This will serve as the major separation control feature for this project, and also enable comparison with experimental data on the matter presented by Lo and Eaton.

3.1.3.2 Induced swirl as an active flow control

A swirl generator is a type of flow control that can be used to increase the overall pressure recovery. Swirl also exists in many applications of diffusers because of

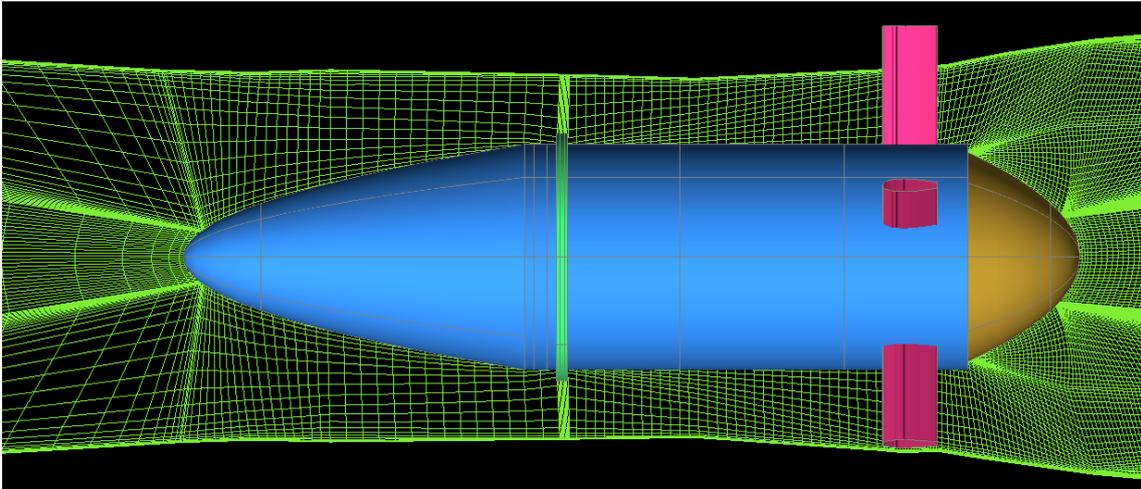


Figure 3.7: A visualisation of the mesh used for $k - \omega$ -SST-SAS simulation. This mesh has about 1800000 cells, making it roughly six times as many as the one used for $k - \omega$ and $k - \omega$ -SST. Noteworthy is how this mesh has a significantly higher concentration of cells close to the walls than the one in figure 3.6, as well as the fact that the number of cells in the area right after the end of the center body is very high compared to other areas of the same mesh.

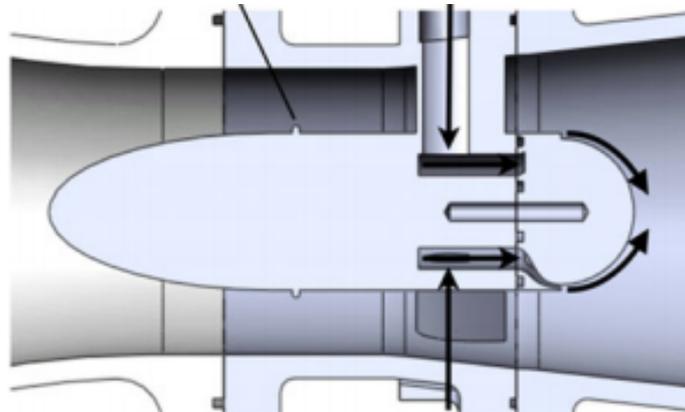


Figure 3.8: A visualisation of how the Coanda effect is implemented as an active flow control to achieve control over the separating wake. The arrows shows how the injected water, by jets surrounding the end piece, follows curvature of the end and consequently removes the wake. Credits for the image goes to Eaton and Lo [2]. Used with permission

preceding components e.g compressors that gives the flow a rotating motion[10]. Swirl was implemented at the inlet of the model in OpenFOAM as an additional boundary condition. It was created as script, the code can be found in appendix A.3. Credits for script goes to Ardalan Javadi.

3.2 Simulation using OpenFOAM

The second major step, after the mesh generation, was to perform the numerical simulation of the flow. The software used to accomplish that was OpenFOAM, a free open-source CFD solver. More specifically, the version FOAM-extend-3.2 was used.

3.2.1 Choice and implementation of solver

The following subsection presents a short description of the implementation of the chosen solver PIMPLE which has been further described in section 2.3.3.1.

3.2.1.1 Values of initial conditions

For this case, the fluid considered is water (which makes the assumption of an incompressible fluid valid). The initial values of the variables used at the different patches were chosen to match this assumption. Furthermore, each variable also possessed a "type", defining the behaviour at patches. The ones used for kOmega/kOmega-SST and kOmega-SST-SAS respectively are presented in table A.2 in appendix A.2.

3.2.2 Strengths and weaknesses of the implemented turbulence models

Choosing which turbulence model to use is one of the more important choices one makes running a CFD-simulation. Different models range vastly in the ability to simulate different physical phenomena. However, the more accurate a model is at capturing the real physics, the costlier in terms of resources such as computing power it usually is. The three different aforementioned turbulence models that was used in this thesis has to be able to simulate complex phenomenon such as separation with a high degree of accuracy. Some of the known strengths and weaknesses of these models according to [11] are summarised below:

- *k- ω* : High accuracy for boundary layers with adverse pressure gradient. Can be easily integrated into the viscous sub-layer without any additional damping functions. Accurate for free shear flows and separated flows. Suffers from weaknesses when applied to flows with free-stream boundaries.
- *k- ω -SST*: Good at predicting boundary layers under adverse pressure gradient flows. Usually performs well for cases regarding turbomachinery.
- *k- ω -SST-SAS*: The large scale eddies are fully resolved, this is not the case

for $k-\omega$ and $k-\omega$ -SST and it grants a more accurate result. The disadvantages of using LES are high cost in computing power and requires a more refined mesh.

3.3 Post-processing

According to the aim presented in the introduction section, the goal is to produce a three component mean velocity field, which then will be compared to experimental data. Therefore, other variables, such as the turbulent kinetic energy k or the specific dissipation rate ω , will not be subject to presentation in the post-analysis. However, during the simulation, the behaviour of such variables was used to either (in the initial steps of the simulation) detect areas where the mesh had to be refined, and later on use the residuals of them to ensure that the solution had converged.

Furthermore, the three component mean velocity was achieved by first running each simulation for $10-30s$ (the time limit chosen in terms of the time frame of the natural process the simulation aims to mimic). Then, a mean value feature was added enabling a time averaged velocity field to be formed over 2-4 seconds. Such time interval was thought of to be well enough for one particle to travel from the inlet to the outlet.

Once the simulations were finished, all creation of relevant figures of the velocity fields were done using Paraview. Noteworthy here is that the scales and actual colours in the velocity profile figures was chosen to mimic the visualisation used by Lo and Eaton, as the figures later on will be subject to comparison with theirs.

4

Results

In order to simplify the presentation, the results are divided into cases depending on the turbulence model and flow control method used. Additionally, the effect of swirl at the inlet flow is presented in case XIII. Altogether there are 13 cases, these are listed in table 4.1.

For the following four sections, all cases are divided into five sub-categories, corresponding to a specific setup.

Discussion concerning why the results look the way they do and comparison with experimental data will be handled in the Discussion section.

	Straight section length	Coanda Blow	Swirl	Turbulence models
Case I	0	0	0	$k-\omega$
Case II	0	0	0	$k-\omega$ -SST
Case III	0	0	0	$k-\omega$ -SST-SAS
Case IV	0	1	0	$k-\omega$
Case V	0	1	0	$k-\omega$ -SST
Case VI	0	1	0	$k-\omega$ -SST-SAS
Case VII	$2D_{CB}$	0	0	$k-\omega$
Case VIII	$2D_{CB}$	0	0	$k-\omega$ -SST
Case IX	$2D_{CB}$	0	0	$k-\omega$ -SST-SAS
Case X	$2D_{CB}$	1	0	$k-\omega$
Case XI	$2D_{CB}$	1	0	$k-\omega$ -SST
Case XII	$2D_{CB}$	1	0	$k-\omega$ -SST-SAS
Case XIII	0	0	1	$k-\omega$ -SST

Table 4.1: Cases that has been simulated. The cases are combinations of setups where the presence of a straight section, a Coanda jet and swirl at the inlet has been varied.

4.1 Case I, II & III

Here are results based on the cases where no straight section nor any Coanda jet was implemented presented. Such geometric setup is presented in figure 3.1. As the plots of the residuals imply, the results have converged well in the cases shown in figures 4.1-4.10. This can be seen through looking at the values along the y -axis which are all very low according to what can be seen as valid [13].

Firstly, the computed variable in all the figures 4.1-4.10 is the mean value for the x -component of the flow divided by the bulk velocity, i.e. \bar{U}_x/U_{bulk} . The value $U_{bulk} = 2.26ms^{-1}$ was taken from report from Eaton, Lo and Elkins [2]. Furthermore, all distances are made dimensionless to relate the actual position (initially entered in mm) to the diameter of the center body. For example, as the length of the center body is 4 times its diameter, the start of the back flow area is at 4 units downstream along the x -axis.

For this project the most important physical phenomenon is the separation bubble starting immediately after the centre body. In order to highlight this area and to make it easier to later compare with experimental data, a scale in terms of colour was chosen so that all cells having a value below 0 has the same colour. In that way, a good illustration of the size of this area is made.

To start with, $k - \omega$ (Case I), is presented in figure 4.1. The results shows a very distinct separation bubble in the wake after the center body. The wake is of a fairly constant diameter, which is a little unexpected and therefore subject for further comments in the discussion section.

The $k - \omega$ -SST model (Case II) in figure 4.2 manages to produce a separation bubble, but of a different shape than for Case I. The separated region for this case is smaller, and therefore another topic for discussion as the size does not entirely match experimental data.

Lastly, Case III (figure 4.3) shows the disappointing fact that the $k - \omega$ -SST-SAS model has failed severely. The resulting figure indicates that a large separation region is developing at the outer wall, immediately after the end of the center body. This behaviour is not be expected when there is no separation control. Therefore, this result indicates that either the mesh or the simulation algorithms are incorrect. This problem will be further assessed in the discussion section.

4.2 Case IV, V & VI

The second section of the results describes the same geometrical setup as the previous one, 3.1. The difference is the implementation of a Coanda jet with a blow rate set to 1. Such blow rate means that the boundary condition across the jet patch of the

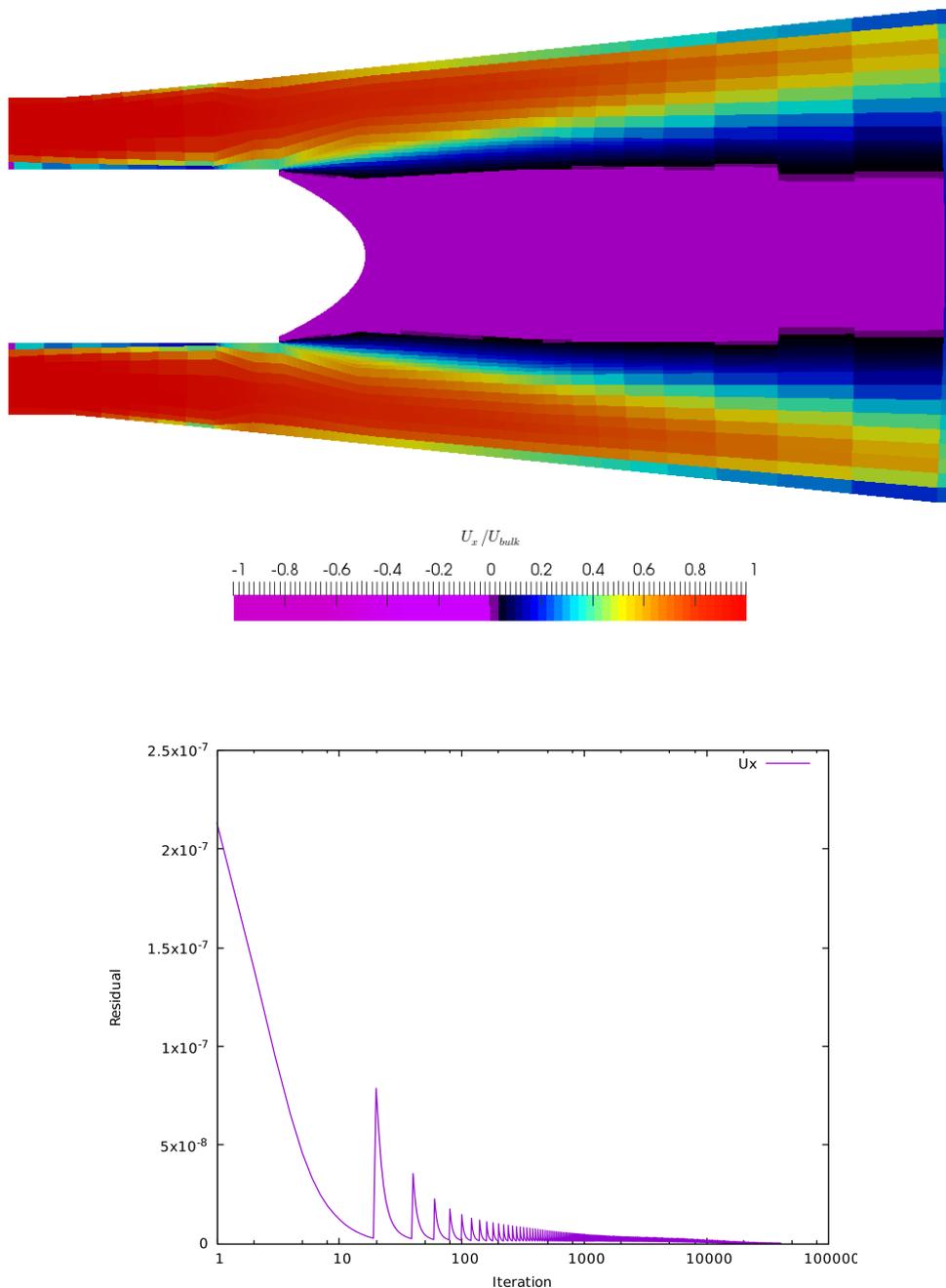


Figure 4.1: Top: a numerical representation of the velocity distribution with $k - \omega$ modelling (Case I). Bottom: plot showing the residuals of U_x , implying that the results have converged.

mesh is the same as the surrounding bulk velocity ($U_{bulk} = 2.26m/s$). The effect and expectation of this is discussed in the theory section.

Attention is now drawn to figure 4.4, showing the numerical representation of case IV. This result shows some interesting features, capturing flow tendencies that are expected in this scenario. Firstly, the obvious fact that the separation bubble reduces

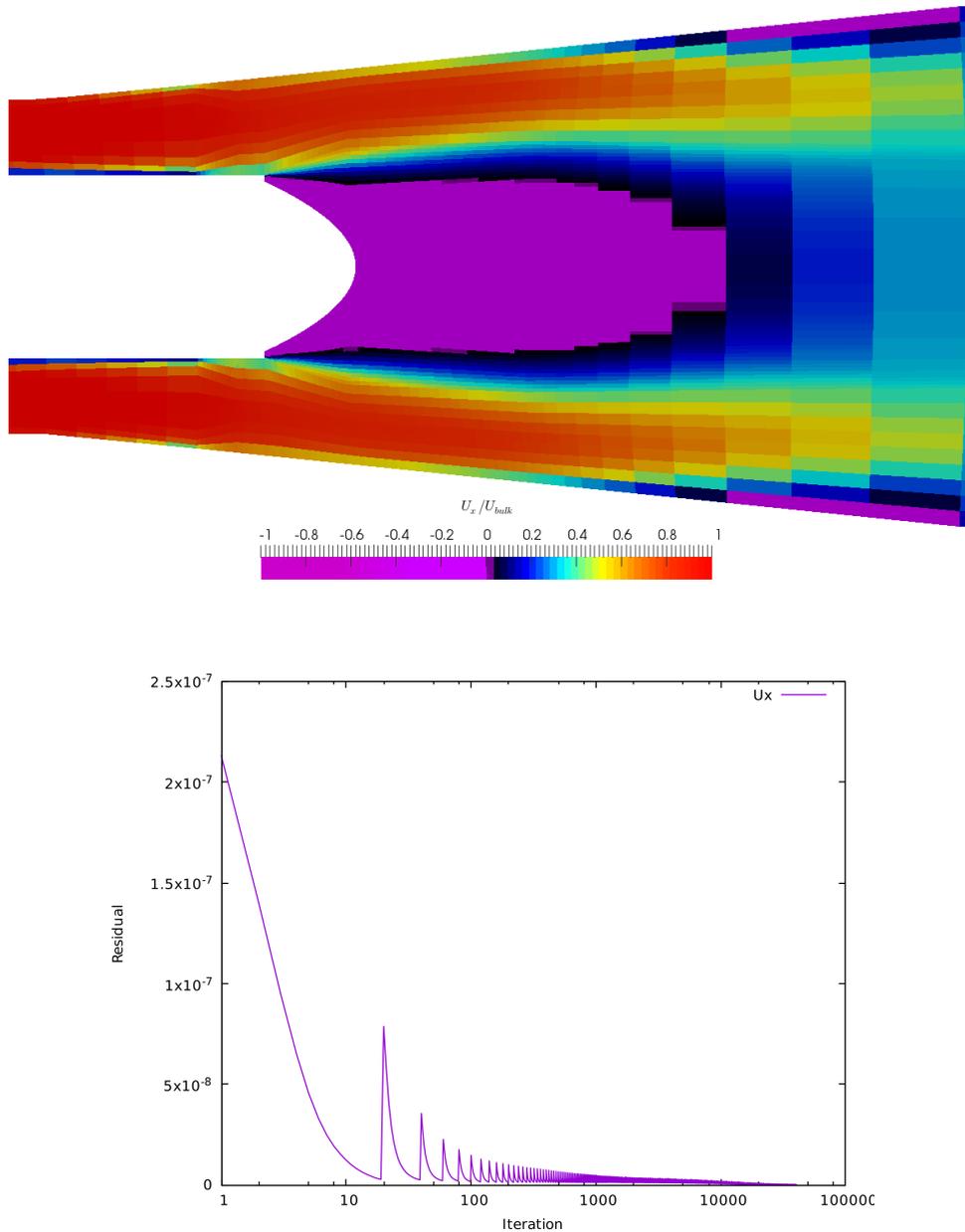


Figure 4.2: Top: a numerical representation of the effect of velocity distribution with $k - \omega$ -SST modelling(Case II). Bottom: plot showing the residuals of U_x , implying that the results have converged.

significantly in size is pointed out. The effect of a Coanda blow in theory as well as in experimental data is to reduce, or even completely close (depending on the blow ratio, geometry and flow) the separation bubble. The bubble in this case is not completely closed, but the reduction indicates a connection between its size and the introduction of a jet. Additionally, the development of a separated region along the outer wall after the closure of the bubble is also in accordance with the wake-wall separation coupling encountered in both theory and experiments.

The results from Case V is presented in figure 4.5, which shows a large separation

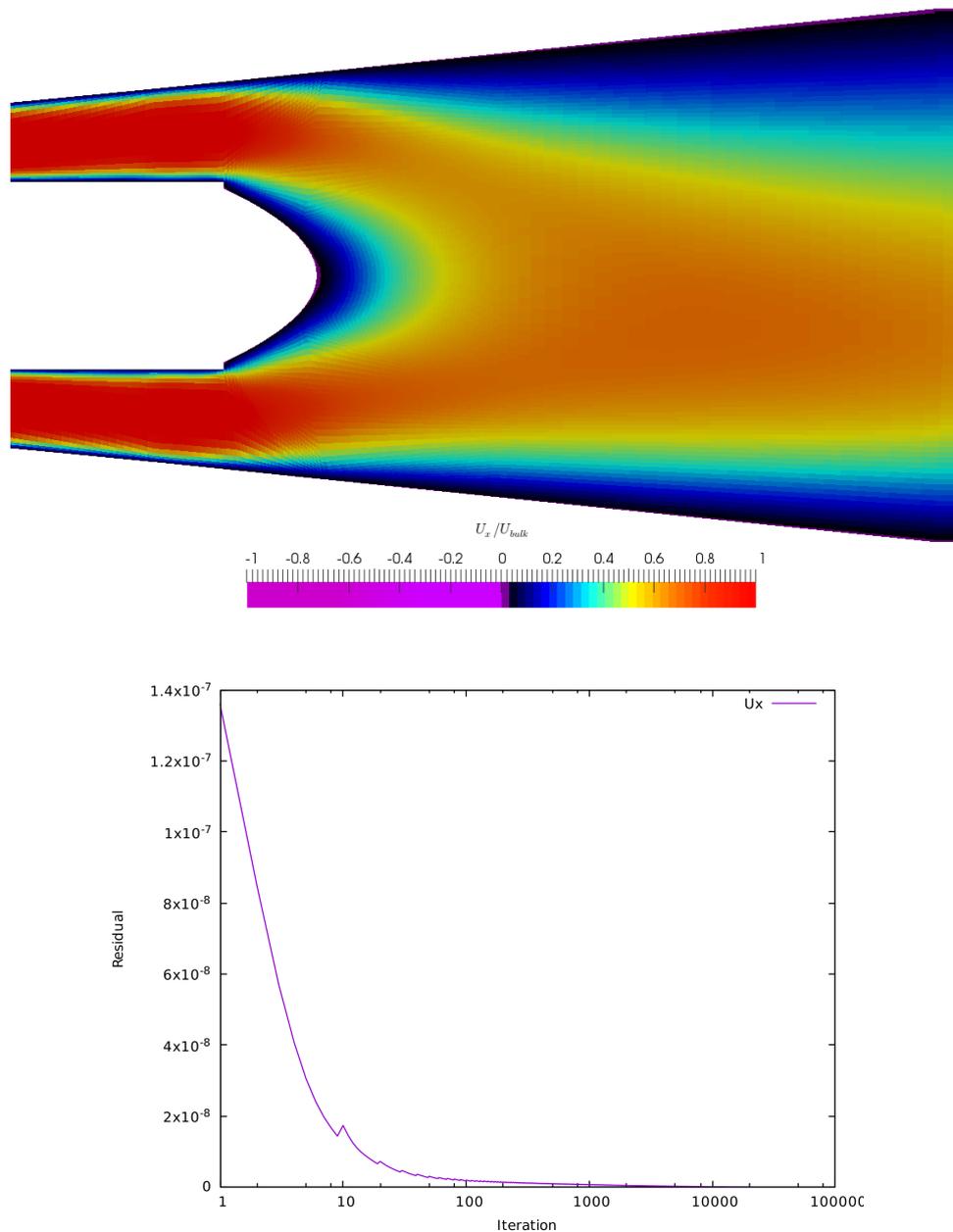


Figure 4.3: Top: a numerical representation of the velocity distribution with $k-\omega$ -SST-SAS modelling (Case III). Bottom: a plot showing the residuals of U_x , implying that the results have converged.

bubble forming after the end of the center body. In comparison, the same model without a jet showed a separation bubble that closed only slightly further out from the start of the bubble 4.2. This model lacks the outer separation region shown by its non-jet counterpart. Hence, the $k-\omega$ -SST model shows poor results compared to the expectations from the experimental data when the jet was implemented.

Figure 4.6 shows a simulation done with the SAS model, now under the impact of a jet. The results are quite similar to what can be expected as the bubble has closed, a stagnation point with low velocity very close to the end of the center body and a

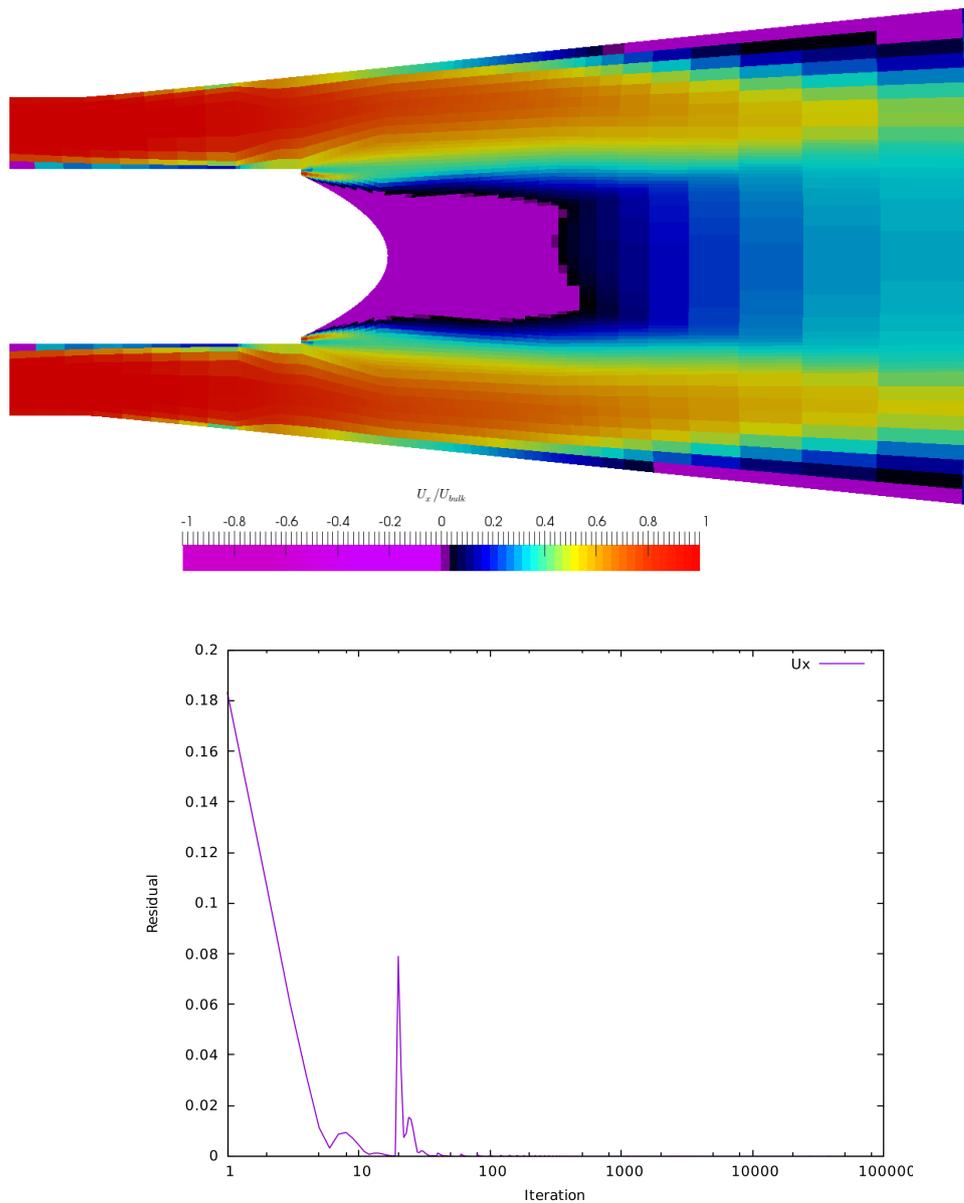


Figure 4.4: Top: a numerical representation of the velocity distribution under the effect from a Coanda jet with $k - \omega$ modelling (Case IV). Bottom: plot showing the residuals of U_x , implying that the results have converged. (final residual= $6.98e-6$)

developing separation region close to the wall. On the other hand, one could argue that the figure 4.3 and 4.6 are too similar for it not to be a coincidence and that the simulation has still not simulated the flow correctly.

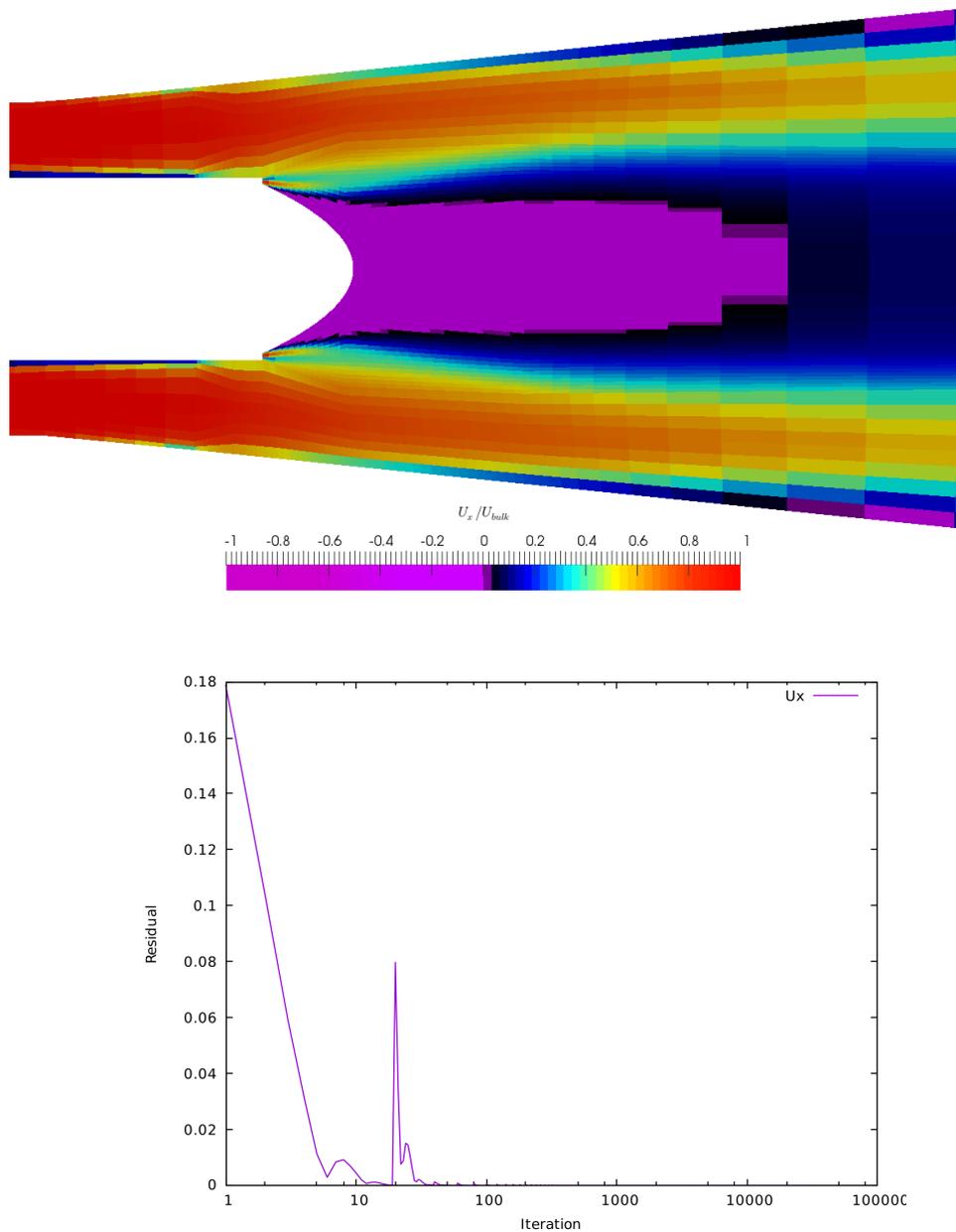


Figure 4.5: Top: a numerical representation of the velocity distribution under the effect from a Coanda jet with $k - \omega$ -SST modelling (Case V). Bottom: a plot showing the residuals of U_x , implying that the results have converged. (Final residual = $8.78e-6$)

4.3 Case VII and VIII

As the two previous sections indicated, the results from the simulations using SAS modelling were poor. Therefore, it was decided to not use this model for the second geometry and instead focus on the previous cases.

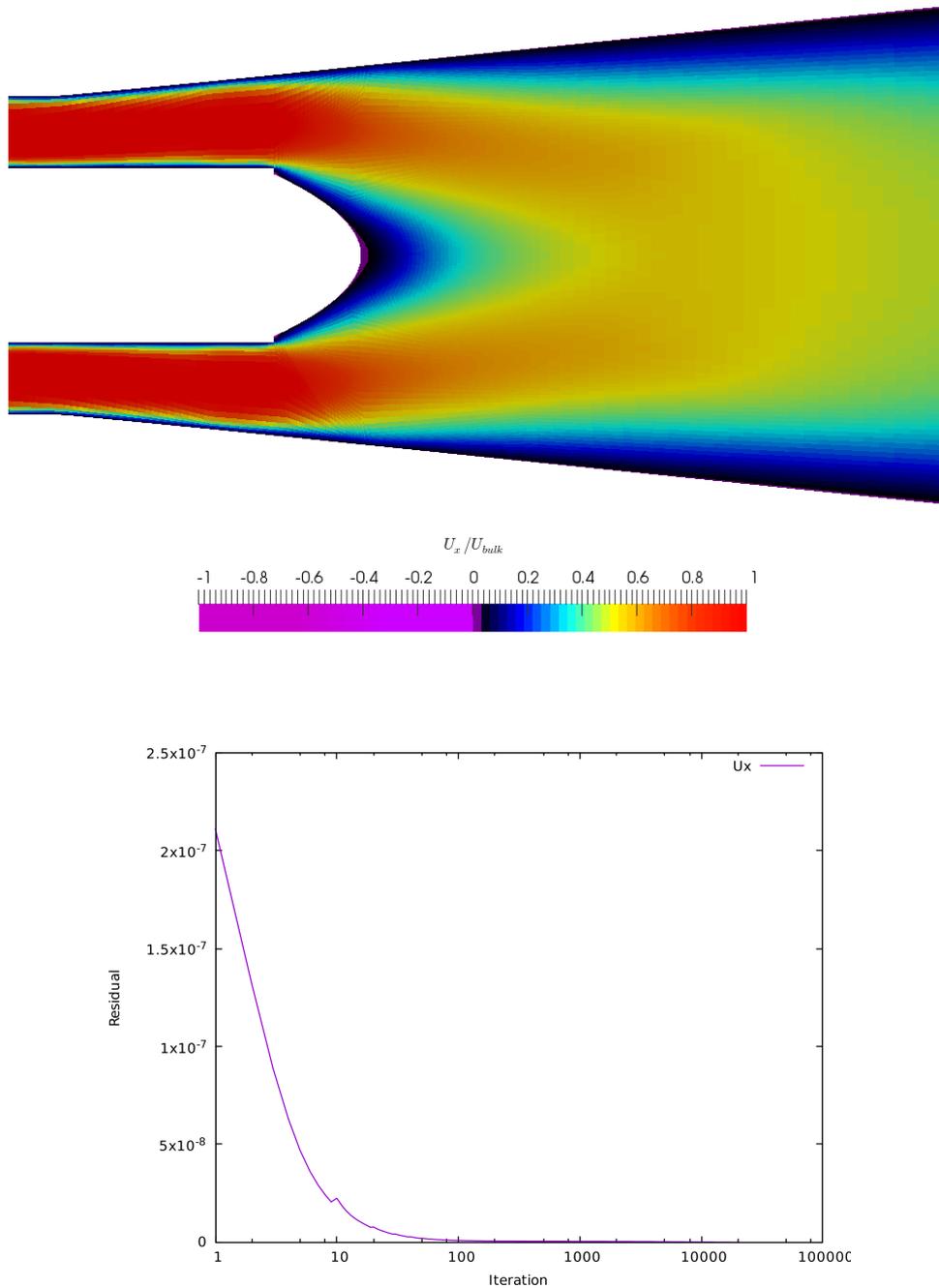


Figure 4.6: Top: a numerical representation of the velocity distribution under the effect from a Coanda jet with $k - \omega$ -SST-SAS modelling (Case VI). Bottom: a plot showing the residuals of U_x , implying that the results have converged.

In the third section of the results, a different geometrical setup was used. The difference between the first and the second geometry was that a straight section was installed in the latter. A view of this setup can be seen in 3.5.

There are, in the absence of a SAS model, two cases to look further into. Figure 4.7 shows how the inserted straight section completely closes the separation bubble for

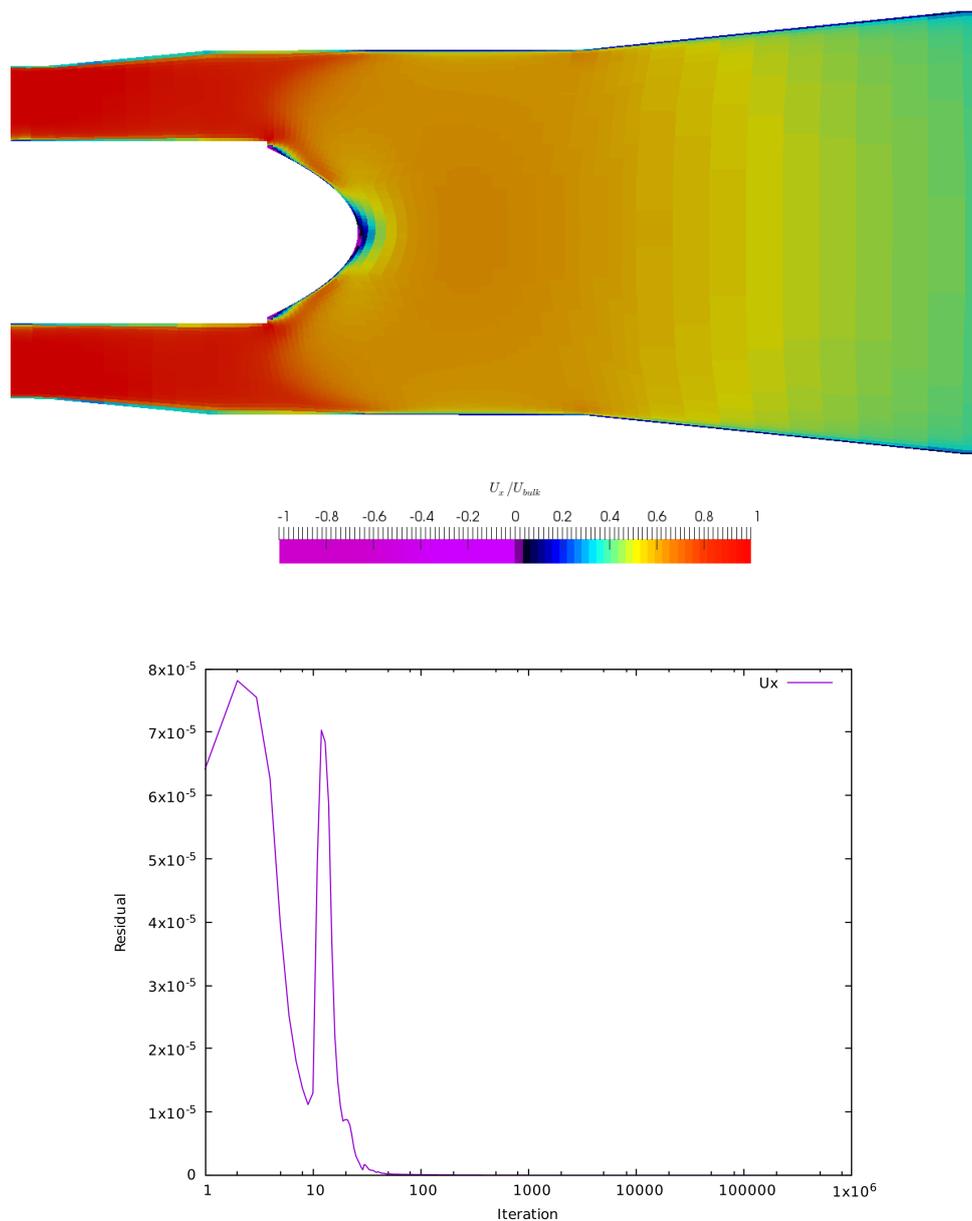


Figure 4.7: Top: a numerical representation of the velocity distribution under the effect from a separation control, in the shape of a straight section, with $k - \omega$ modelling (Case VII). Bottom: a plot showing the residuals of U_x , implying that the results have converged.

the $k - \omega$ case The flow after the center body is almost completely homogeneous in terms of velocity profile in the yz -plane (i.e. for each x value). Something worth noting is the small stagnation region at the very end of the center body, providing the only indication of the separation bubble that, according to experimental data, should be present here.

In summery, the results for the $k - \omega$ model show an overestimation of the effect of the straight section, as a complete closure of the separation bubble only is expected

(once again according to experiment) when a Coanda jet is also present.

Attention is moved to case VIII where results are presented in figure 4.8. The results show a separation bubble, which in size and shape is very similar to experimental data. Additionally, the tendency for the development of a separation region along the wall in the diffuser is present, which also agrees with experimental data.

4.4 Case X, XI

The figures for the cases in figures 4.9 and 4.10 compared to their corresponding simulations without any Coanda jet in figures 4.7 and 4.8 indicates that the flow pattern remained almost constant. Hence, the chosen model and produced mesh seems unable to capture the expected behaviour with the implementation of the Coanda jet.

4.5 Case XIII

For the last case, the concept of swirl at the inlet in order to increase the pressure recovery has been tested. Therefore, the results presented in figure 4.11 and 4.12 also contains a pressure distribution in the domain, as well as the usual velocity field. For a further comparison, the pressure distribution of the same model used but without swirl is presented in figure 4.13.

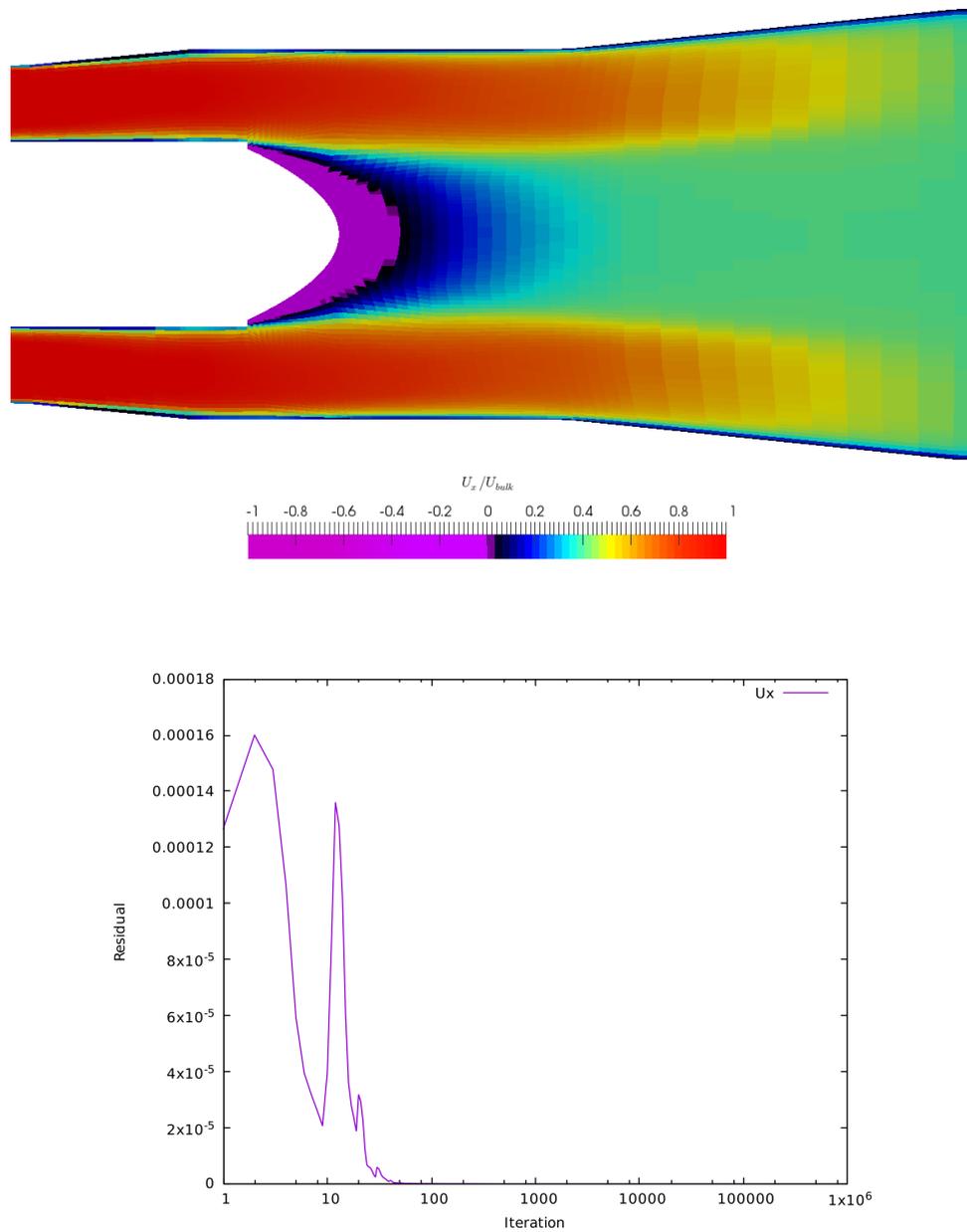


Figure 4.8: Top: a numerical representation of the velocity distribution under the effect from a a straight section separation control mechanism with $k - \omega$ -SST modelling (Case VIII). Bottom: a plot showing the residuals of U_x , implying that the results have converged.

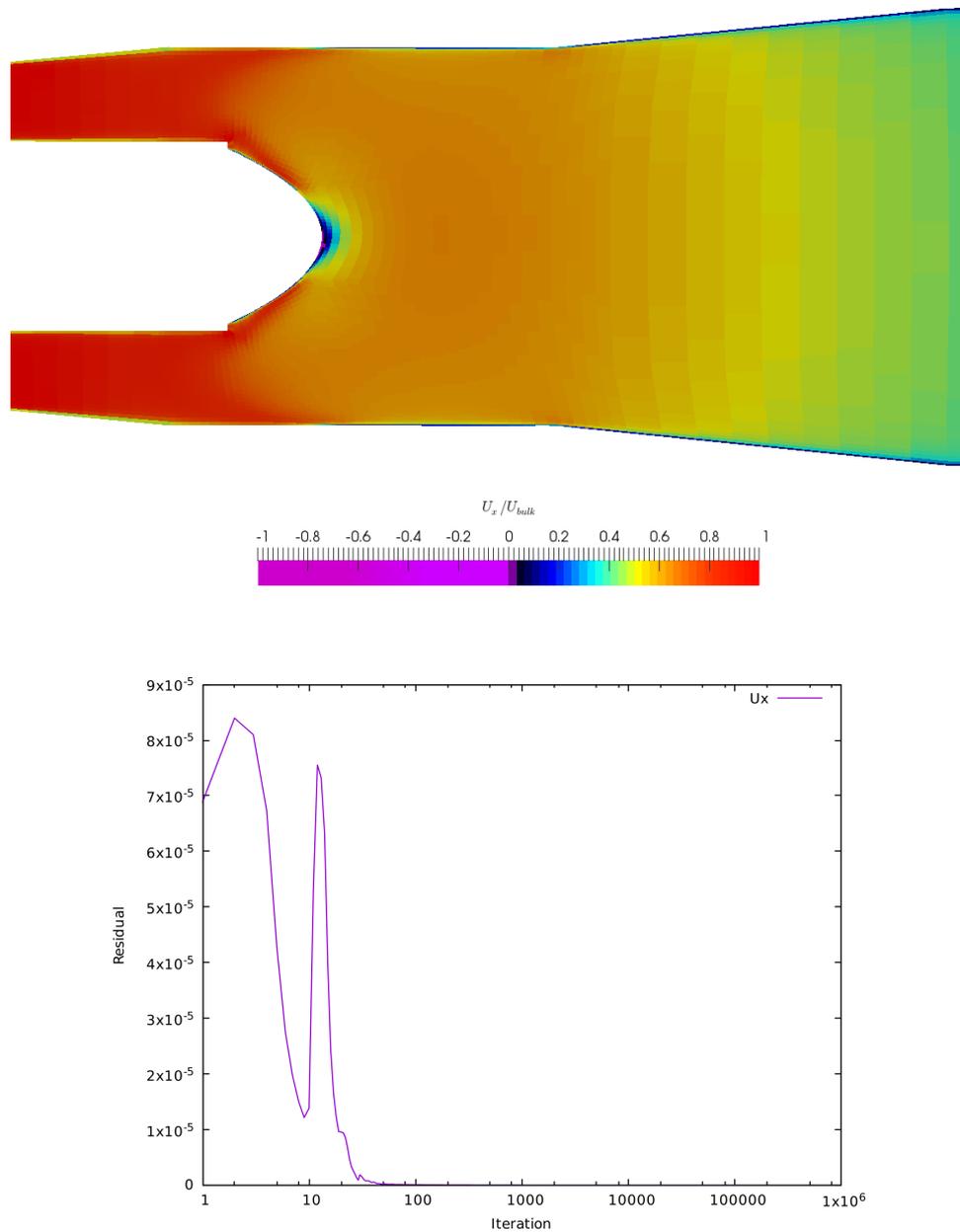


Figure 4.9: Top: a numerical representation of the velocity distribution under the effect of a straight section and with a Coanda jet implemented. The simulations was run with $k - \omega$ modelling (Case X). Bottom: plot showing the residuals of U_x , implying that the results have converged.

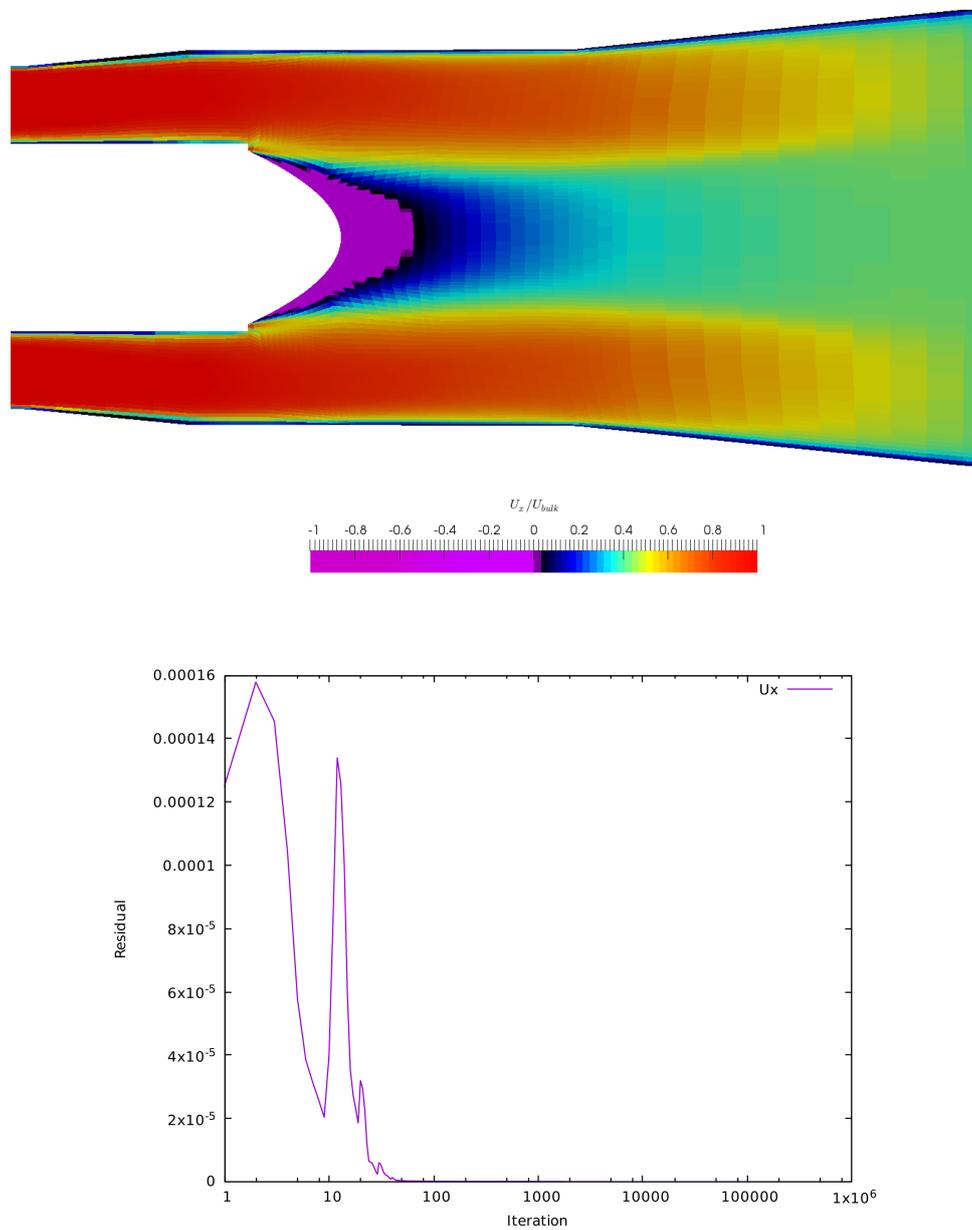


Figure 4.10: Top: a numerical representation of the velocity distribution under the effect of two separation control mechanisms, a straight section and a Coanda jet. The turbulence model chosen was $k - \omega$ -SST (Case XI). Bottom: plot showing the residuals of U_x , implying that the results have converged.

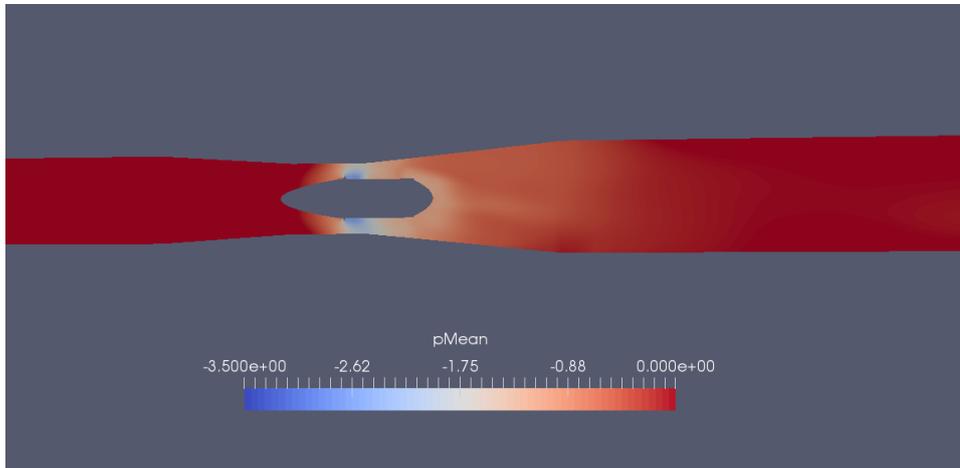


Figure 4.11: Case XIII showing the pressure field with swirl at the inlet. As the pressure recovery was the variable under investigation, the negative to zero scale was chosen to highlight this.

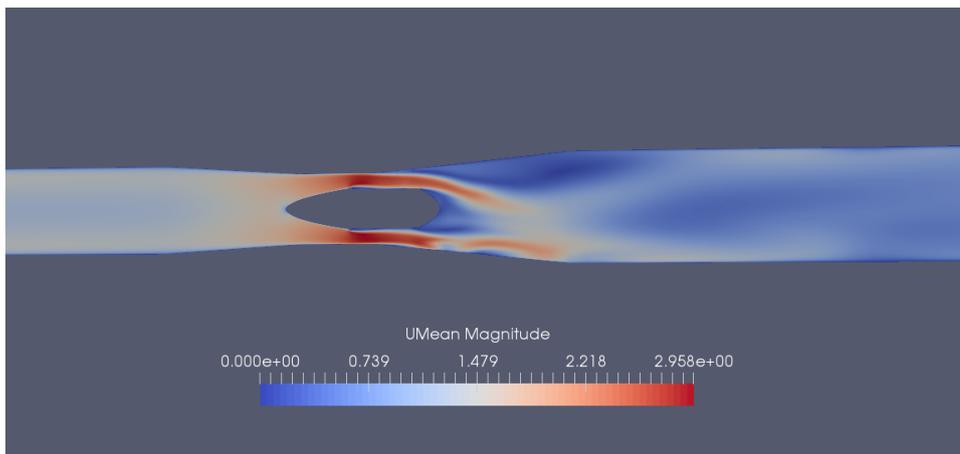


Figure 4.12: Case XIII showing the numerical representation of the velocity distribution for k - ω -SST profile and with swirl at the inlet. The inhomogenous field in the diffuser is further discussed in section 5.5.

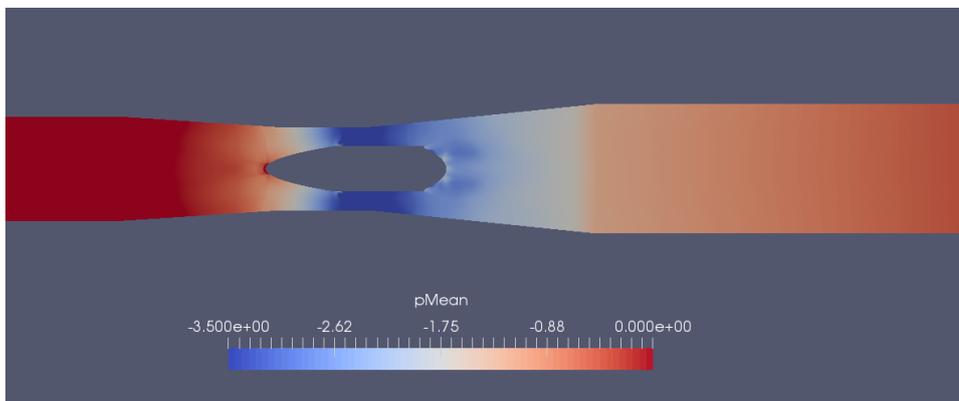


Figure 4.13: Time averaged pressure distribution in the domain for a simulation run using $k-\omega$ -SST modelling. The major concern for further investigation was the pressure recovery, which is the reason for the negative to zero scaling.

5

Discussion

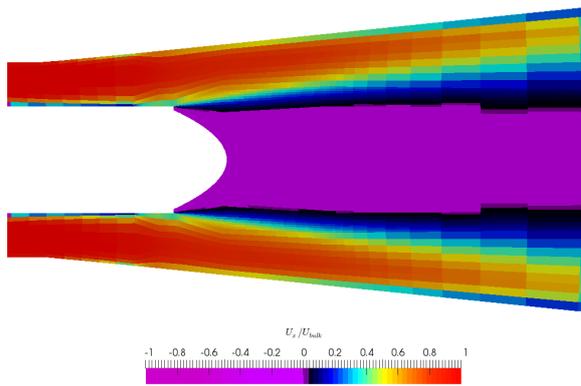
This section will initially focus on the aim of the thesis, evaluating different turbulence models ability to capture a separation bubble and flow control methods used to minimise it. This is mainly done by a comparison between the results from the simulations and experimental data. The experimental data is retrieved from Lo and Eaton's paper on the matter [2], and will from here on be referred to as "the experimental data". Differences and similarities will be pointed out, and consequently a subject for further discussion on why the results look the way they do compared to the experiments. Further experiments and figures might have to be included to highlight certain insights made.

5.1 Comparison between simulations and experimental data without Coanda jet

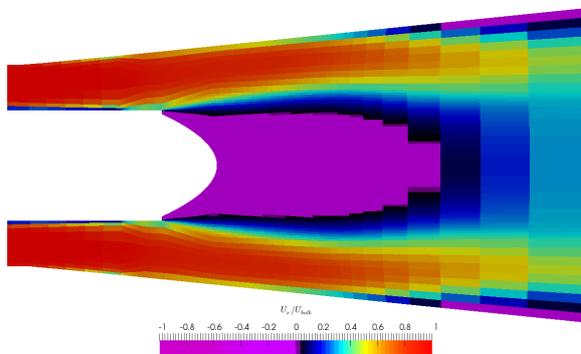
Firstly, the discussion will aim to investigate how well the simulations managed to capture the motion of the flow when no Coanda jet was used. Hence, this section covers a geometry with and without a straight section, divided into the two following subsections. In the experimental data, the addition of a straight section caused a major decrease in size of the separation bubble in the wake after the center body; therefore, the initial discussion will be based on visual observations of sizes and patterns, rather than quantified measurements as a closed separation bubble or not can simply be observed.

5.1.1 Cases I, II & III (no straight section)

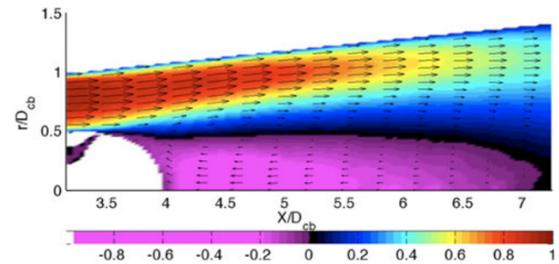
For case I, II and III, the experimental data showed a large separation bubble in the wake after the center body, as seen in figure 5.1c. The physics behind such development in flow pattern is described in section 2.5, and can be said to be caused by the presence of an adverse pressure gradient. The region of separation will serve as the most important comparison tool in this section.



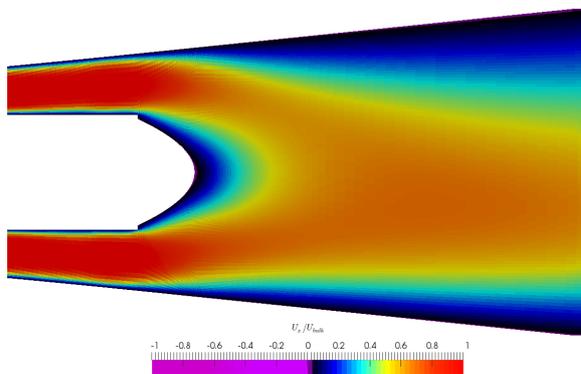
(a) $k-\omega$ model



(b) $k-\omega$ -SST model



(c) Experimental data



(d) $k-\omega$ -SST-SAS model

Figure 5.1: A comparison between three turbulence model and experimental data. Courtesy to Lo and Eaton [2] for the figure to the right of the experimental data. Used with permission

To start with, the result of Case I $k-\omega$, appears to take a quite similar shape as the experimental data. The visual comparison is done by comparing figure 5.1a to figure 5.1c. As described in the previous section, the length agrees very well with the experimental data. The shape however is not very similar, especially as the distance

from the end increases. According to the theory, $k - \omega$ is expected to perform worse than the two other models, and should specifically have problems in areas with an adverse pressure gradient. Therefore, it possible to state that the results are good and do to a many extents capture the general motion of the fluid, especially relative to what was expected.

$k - \omega$ -SST however was expected to perform better in this area, as one of the great benefits with the addition of the SST-term is the ability to better capture separation in an adverse pressure gradient area, i.e. the situation at hand. As a comparison between figures 5.1b and 5.1c indicates, the phenomenon of interest for those cases is without a doubt separation. With that said, the evaluation of this turbulent model is twofold. On one hand, $k - \omega$ -SST managed to replicate the shape of the separation bubble in a much better manner than $k - \omega$. Since the mesh used for the two was the same, this must be due to modelling advantages rather than originating from a finer mesh. On the other hand, the size, and more specifically the length, of the separation bubble is about half of the experimental data.

A possible reason to why the separation bubble closes earlier is a concept further explained in section 5.3 and concerns the coupling between the inner separation in the wake and the outer separation at the wall. In the case with $k - \omega$ -SST, the too sudden end of the central separation bubble is immediately followed by the development of an outer separation region along the diffuser wall. It might be that the reason the central separation bubble closes is that this concept goes the other way around, i.e. numerical problems causes the flow to separate at the outer wall and therefore closes the bubble prematurely. If so, there must be an underlying difference between $k - \omega$ and $k - \omega$ -SST.

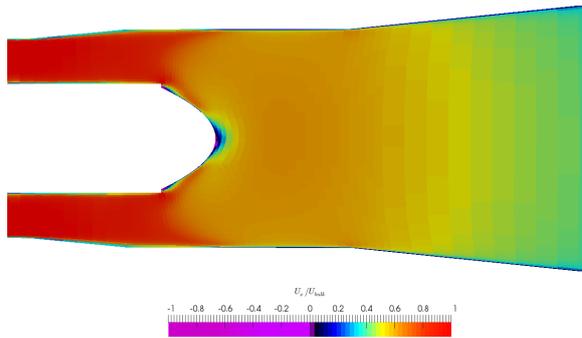
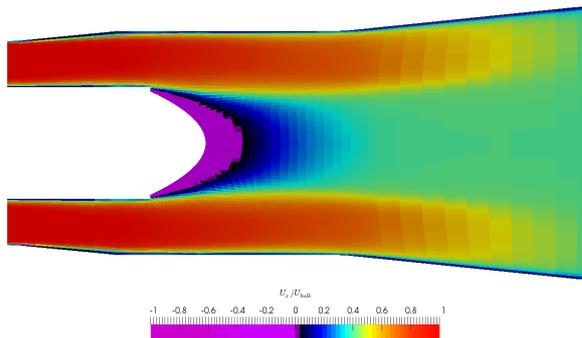
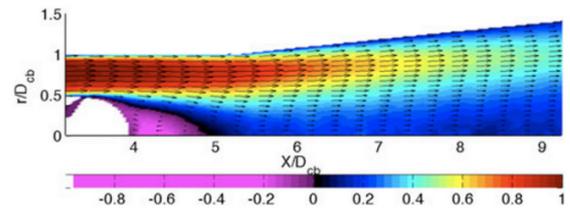
$k - \omega$ -SST is fundamentally speaking a hybrid model, behaving like $k - \omega$ or $k - \epsilon$ depending on the situation. Therefore, it could be that the pace of the switch from a $k - \epsilon$ behaviour in the free stream to $k - \omega$ close to the wall is insufficient. In other words, $k - \omega$ -SST's inability to correctly capture the size of the separation bubble in contrast to $k - \omega$ models could originate from the chain of events where a mesh with inadequate y^+ for this model causes the flow to be modelled with $k - \epsilon$ in a too large region and that triggers a wall separation stronger than the wake separation, upon which the former forces the latter to close prematurely.

Lastly, attention is turned towards the SAS model. No longer comparison is needed to conclude that the implementation of this model must consist of some fundamental problems, as visualised in figure 5.1d and 5.1c. As later sections will indicate, the poor results from the SAS model were recurring. Therefore, a section dedicated solely to a discussion on the matter is presented in section 5.3.

5.1.2 Cases VII, VIII, IX (with straight section)

A straight section is introduced in order to evaluate if it is possible to close the separation bubble without the means of active flow control (which needs external

energy). The drawback of the straight section is that it results in a longer diffuser, which results in a more costly diffuser taking up more space. Nevertheless it was seen as a rather effective way to minimise the separation bubble in the experimental data and is therefore included in this thesis. Experimental data shows that the separation bubble still exists but closes before the start of the diffuser section, as illustrated in figure 5.2c.

(a) $k-\omega$ model(b) $k-\omega$ -SST model

(c) Experimental data

Figure 5.2: A comparison between two turbulence models and experimental data under the impact of passive separation control, implemented through an installed straight section. Courtesy to Lo and Eaton [2] for the figure to the right of the experimental data. Used with permission

Starting with the $k-\omega$ modelling, the separation bubble in figure 5.2a was totally closed which compared to the experimental data is not to be expected. According to Eaton, Lo and Elkins [19], the reason behind why the bubble closes naturally with a straight section is that it defers the onset of the adverse pressure gradient.

Comparing the $k-\omega$ -SST modelling in figure 5.2b to figure 5.2c, the results was almost identical to the experimental data. The $k-\omega$ -SST model gave a better result than $k-\omega$, which was expected because the SST-model has the ability to better capture separation (this was discussed in section 2.2.1.2).

The $k-\omega$ -SST-SAS model was excluded from comparison with the experimental

data because of poor results from simulations for the mesh constructed for running SAS with geometry 1. The reason for why the quality of the generated mesh was low will later be discussed in section 5.3.

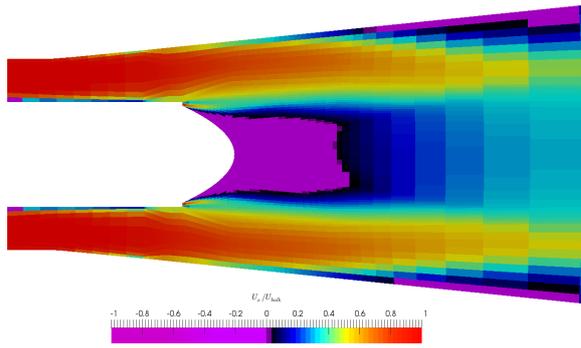
5.2 Discussion of results from simulations with Coanda blowing

Coanda blowing is a type of flow control that was used in the cases that will be discussed in this section. Results of the numerical simulations showed the effect of the Coanda blowing to be almost non-existent. The Coanda effect was used in simulations with and without a straight section. Below, the Coanda effect will be discussed for both variants.

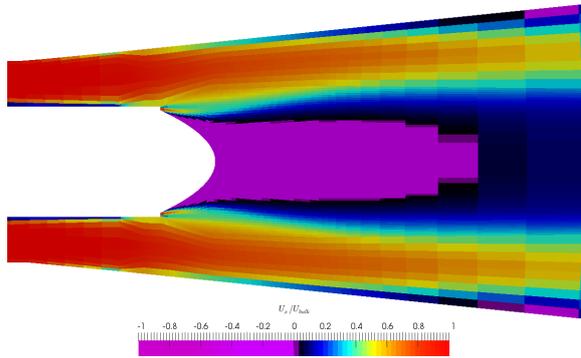
As mentioned earlier in the theory section (section 2.5.2), the Coanda effect is the tendency for a fluid to follow the curved surface. The Coanda effect used as a separation control feature is realised here by the fact that fluid from the jet follow the end piece and eventually inhibits the development of a separation wake by injecting a stream of fluid with the same, approximate, velocity as the surrounding fluid.

Firstly, it is safe to conclude that the three turbulence models used gave very different results. Secondly, a relevant comparison is to see how the models react to the jet in terms of their look in figure 5.3 compared to 5.1. The only model of the three that shows a behaviour close to that of experimental data is $k - \omega$. For $k - \omega$ -SST, the recirculation region grows in size and for $k - \omega$ -SST-SAS, the addition of the jet does not cause any difference, as figures 4.3 and 4.6 are quite similar. Since the result from simulations where the $k - \omega$ -SST-SAS model is used again appears to be invalid, it is from this point excluded from comparisons with the other two, and instead is dedicated a section (section 5.3) discussing the failure of the model further.

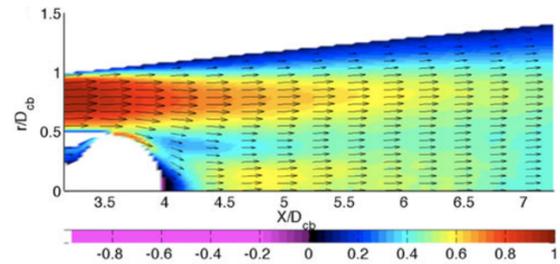
The most probable causes for the difference in result between simulations where Coanda blowing and experimental data ones will now be discussed. The main suspect cause is low mesh quality, especially around the region near the end part. Bad mesh could be in the form of the mesh being too non-orthogonal, too sparse or containing too big "jumps" in the mesh. Too big "jumps" refers to that the cell size changes too drastically between two cells. This could result in difficulties when it comes to ensure that quantities are conserved in the governing equations. A further concern regarding mesh quality is if the value of y^+ was in the correct range. However, according to one source [20], $k - \omega$ has a flexible wall function that should be able to model y^+ values both in the viscous, buffer and logarithmic region with acceptable quality. Still, it could be the case that the use of a wallfunction in combination with separation is a cause for problems. A refined version of the mesh used for the $k - \omega$ and $k - \omega$ -SST models was constructed and the effect of that is discussed in section 5.4.



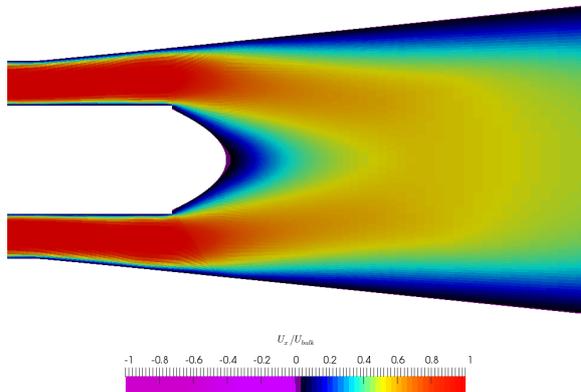
(a) $k-\omega$ model



(b) $k-\omega$ -SST model



(c) Experimental data



(d) $k-\omega$ -SST-SAS model

Figure 5.3: A comparison between three turbulence models with experimental data under the impact of active separation control, implemented through a Coanda blow with a blow rate of 1. Courtesy to Lo and Eaton [2] for the figure to the right of the experimental data. Used with permission

One consideration is the possibility that $k-\omega$ is not a well suited turbulence model for capturing the Coanda effect. The $k-\omega$ -SST model however should have been able to capture it as it has been observed to do in other cases [21].

Lastly, a factor that also could have impacted the result of the Coanda effect is the fact that the boundary condition for the jet was that of a uniform velocity in the x -direction. In retrospect, a more realistic boundary condition would have been to use a fully developed velocity profile as the jet fluid had travelled in a tube before being sprayed out in the domain. This implementation was not tested due to time constraints.

5.3 Discussion regarding the results from the SAS-simulations

Based on previous results, it seems like the SAS simulations are not able to reproduce the flow pattern from the experiments. For example, figure 5.1 shows clearly how the SAS model fails to capture the separation bubble in the wake found in the experimental data. This could be attributed to by many factors, where bad mesh quality is among the most probable.

$k - \omega$ -SST-SAS is a hybrid turbulence model that consists of both URANS and LES features. Specifically, in a boundary layer close to the wall, the flow behaves according to a LES model; which requires a very fine mesh to resolve small turbulent scales at short distances from the wall. A fine mesh is accomplished in many ways. In particular, the mesh should be as isotropic as possible [14], meaning that the aspect ratio must be low, and simultaneously fulfil $y^+ \approx 1$. An retroactive investigation shows that the mesh used for the SAS simulations had the aspect ratio for the cells at the outer wall outside the end of the center body had values from 400 up just below 1000.

Furthermore, another reason for the poor performance of the SAS model, coupled with the aspect ratio, could be that the cell sizes (in all directions) must increase or decrease very gradually. Figure 3.7 shows a 2D plane of the mesh used for the SAS model. Even though the jumps between the blocks are much smaller for this mesh compared to figure 3.6 (showing the mesh used for $k - \omega$ and $k - \omega$ -SST), there are still areas that could create a loss in accuracy. For example, a sudden increase in cell size between adjoining cells is problematic for conservation laws to be correctly implemented. The most defined jumps are found at the nose, at the O-ring and at the end.

To further try to explain the reason as to why the SAS simulations did not meet the expectations, a poor mesh could contribute to the tendency of a separation development at the outer wall, as illustrated in figure 4.3. One possible factor to its different behaviour compared to simulations using the other models (Case I and case II) is that the inferior mesh quality could have made SAS overestimating the outer wall boundary layer, which in return had a closing effect on the separation bubble. Such coupling between the two separation regions is coherent with Eatons dissertation [2] which states that as the separation bubble in the wake closes, there

is an expected development of a separation region at the outer wall. Therefore, if the mesh causes an overestimation of parameters influencing on the separation at the wall, that separation might counteract the tendency for the flow to separate after the center body ends. In other words, as simultaneous separation in the wake and at the other wall (for the same x coordinate) seems unlikely, a mesh triggered (incorrect) development of the latter counteracts the (correct) development of the former.

5.4 Effects on results from improvement of mesh

As previous sections have implied, the mesh quality seem to be of great significance. Therefore, in an attempt to get better results, some mesh improvements were done on the mesh used for the $k - \omega$ and $k - \omega$ -SST models as simulations using those had generated results most closely in resemblance to experimental data. Firstly, the airfoils were omitted to get a more simple geometry to work with. This was done because the number of airfoils were 5, while the usage of an O-grid gave a shape based on 4 sides throughout the domain. This discrepancy caused large complications when generating the mesh. More nodes were added to sensitive areas (e.g. the end and nose sections of the center body) in hope to increase the accuracy of the results.

Furthermore, another alteration made was that the simulation time was increased up to 45 seconds instead of 12 seconds in order to investigate if the earlier simulations were ended prematurely. At last, the jet velocity was, after an initial run at the previously used 2,26 m/s to first 5 m/s and then 10 m/s. This was done to see if an exaggerated Coanda jet affected the velocity profile.

5.4.1 Comparison between old and improved mesh

As seen in figures 5.4 and 5.5, more nodes were added to the more sensitive areas, that could affect the results such as the nose, end and middle section. When the results from the new mesh was compared to the experimental data very sparse improvements, compared to the old mesh, could be seen. Instead, the results were very similar for simulations both with and without the Coanda effect, indicating that the alterations made were not enough.

5.4.2 The effect of increased jet velocity

Another way to look further into the connection between the poor results from the jet simulations and the hypothesis that the model and /or mesh were unable to capture the Coanda effect is to vary the blow ratio of the jet. With a higher blow ratio, it turns out that the Coanda effect can be "forced" to occur. As illustrated in

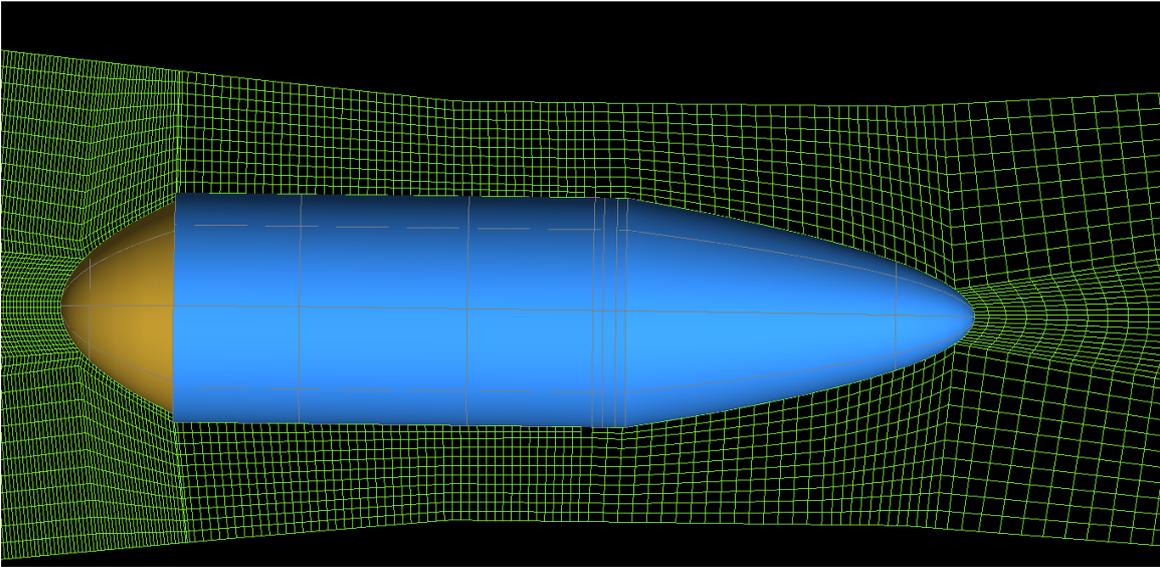


Figure 5.4: Improved mesh

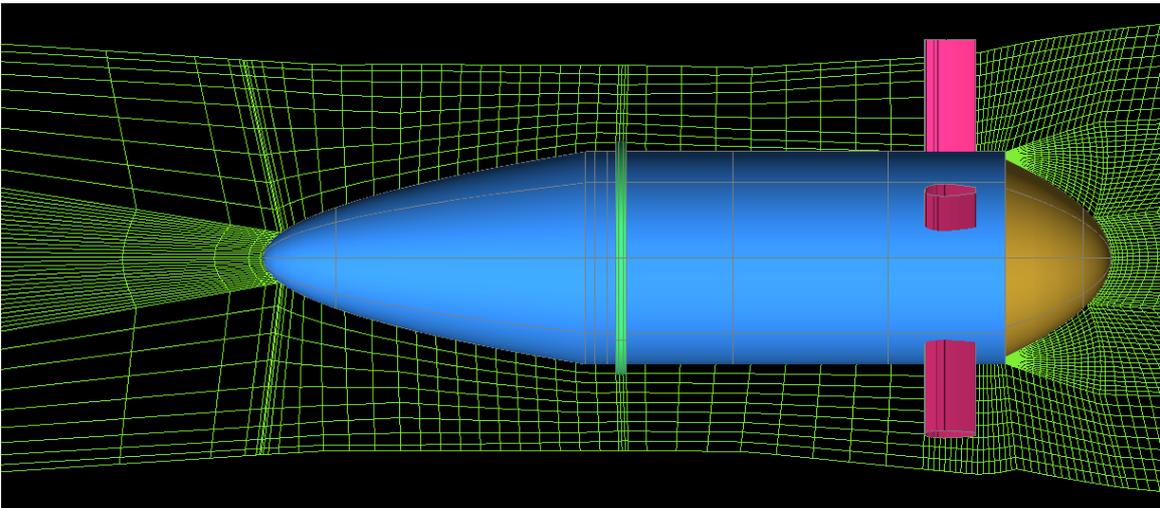


Figure 5.5: Old mesh

figures 5.6, 5.7 and 5.8, a higher velocity jet results in a more pronounced Coanda effect, that eventually almost closes the gap at 10 m/s (corresponding to a blow rate of ≈ 4.4). Therefore, it seems as if the model used underestimates general tendencies in the flow, and only when they are greatly exaggerated, the flow behaves like the experiments implies. The reason is probably, once again, that the mesh quality is insufficient.

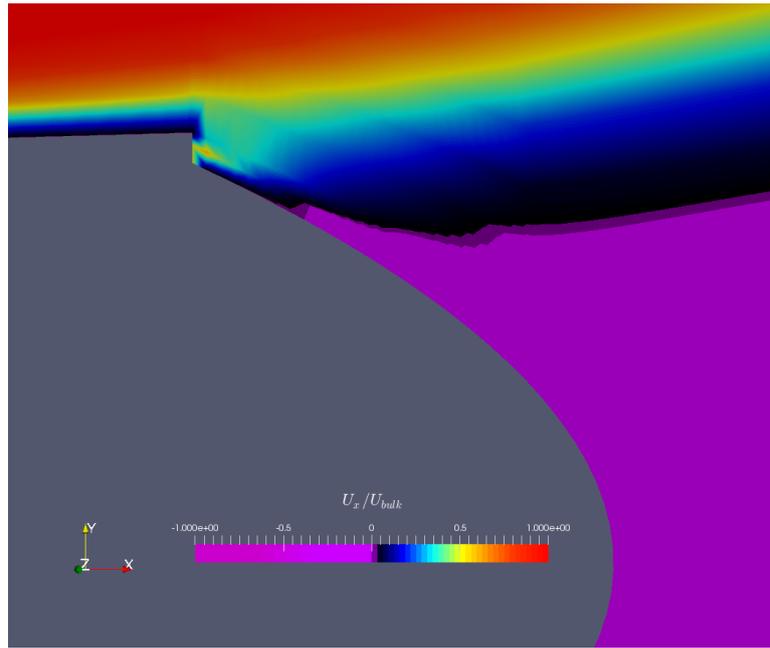


Figure 5.6: Close up picture on the jet/end region just behind the center body. Illustrates how the injected stream from the jet is unable to, according to the Coanda effect, follow the curvature of the end piece.

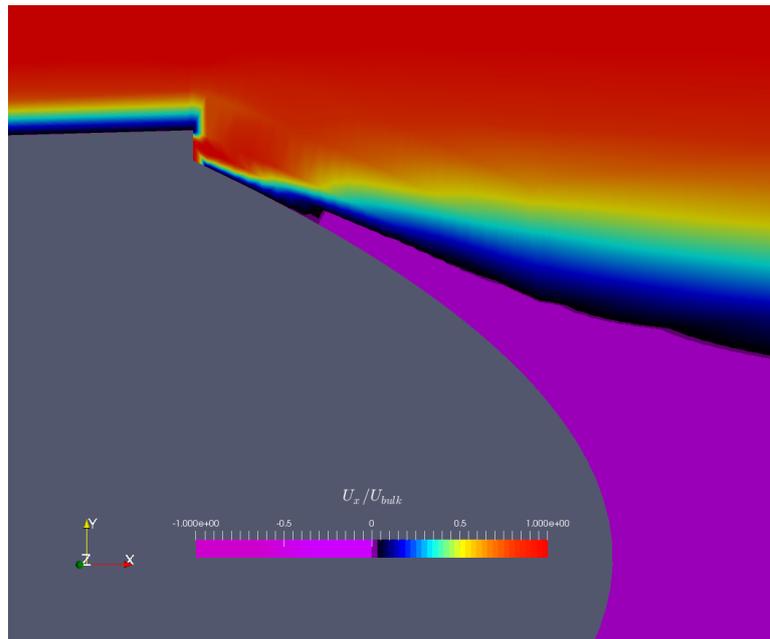


Figure 5.7: The attempt to better capture the Coanda effect by increasing the Coanda blow from 2.26 m/s to 5 m/s, making the blow ratio approximately 2.2

5.5 The effect of swirl on pressure recovery

Lastly, an investigation on the proposed, positive, effect (by Kumar and Kumar [10]) of swirl at the inlet on the pressure recovery was carried out. The implementation

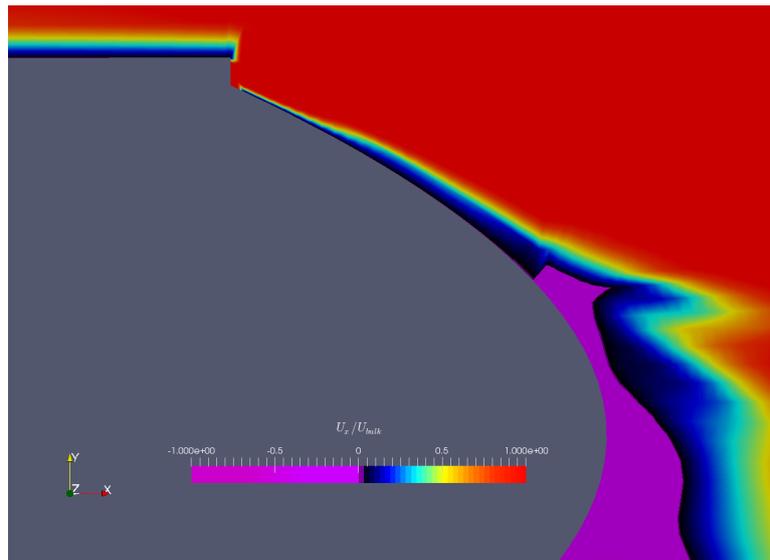


Figure 5.8: Heavily exaggerated Coanda blow rate (10 m/s giving a blow rate of approximately 4.4) manages to capture the Coanda effect along the tail piece. Additionally, the difficulty with using an O-grid to deal with the annular inlet is highlighted by the sudden jump in the separation region towards the end of the end piece.

of swirl was done via a tangential component in the inlet velocity field, causing the flow to not only propagate along the x-axis, but also rotate with an angular velocity.

The simulation with swirl was done using the $k-\omega$ -SST model and geometry 1. The results are presented in figure 4.12 and 4.11 for the velocity and pressure respectively. Additionally, the results from such simulation without swirl, i.e. Case II, is added in figure 4.13 for comparison.

The comparison between figure 4.11 and 4.13 shows that the former has a significantly better pressure recovery. Especially, the separation region after the center body shows a larger drop in pressure when no swirl is present. Hence, the fact that swirl improves pressure recovery is a feature that the created mesh and chosen turbulence model is able to capture.

It is worth mentioning that the velocity profile when swirl is present becomes much less uniform compared to the simulation run without swirl, as indicated in figure 4.12 (with swirl) in contrast with figure 4.2 for a non-swirling inlet velocity field. Such chaotic flow is of course problematic for real usage.

6

Conclusion

The aim of this study was largely divided into three different areas: the physics of the fluid, the numerical simulation, and a comparison of the results from the CFD-simulations to experimental data. Results from simulations without means of active flow control showed close similarity with experimental results (excluding the SAS simulation). The Coanda effect, however, proved difficult to capture in numerical simulation; even if some effect could be shown, it was not as pronounced as in the experimental results. Passive flow control in the form of adding a straight section after the center body proved easier to simulate numerically, although still not completely accurate. The reasons for the numerical simulation's discrepancy from the experimental result has been investigated, and even though not completely clear, a few likely suspects as to why has been discussed. Mesh quality was suspected to be the primary flaw and a new improved mesh was constructed (with airfoils omitted making the creation of a more robust mesh easier) and implemented in OpenFOAM. The results from the new mesh was somewhat improved as the Coanda jet followed the curvature of the end of the center body more closely. Though, it still separated too early, making the effect on the separation bubble insufficient compared to experimental data. Anyhow, there is reason to believe that more flaws exist, either in the mesh or the implementation in OpenFOAM. The lack of experience in the software used in this project coupled with restraints in time resources made troubleshooting a challenge.

Even if the results from the simulations are not fully accurate they still often show tendencies in accordance with experimental cases. All the simulations show a coupling between the development of the wall boundary layer and the formation of a separation bubble after the center body. Simulations that show thick wall boundary layer show a smaller/eliminated bubble and vice versa. Finally the effect of swirl at the inlet showed increased pressure recovery as expected, but at the cost of a less uniform flow.

6.1 Future Work

Something to take into consideration in future work is to omit the airfoils. As stated earlier in the paper, a mesh was in late stages created with the exclusion of the airfoils in order to avoid complex associations and blocking, possibly risking a bad mesh as a result. A mesh setup without airfoils for each respective case could be made in order to get a mesh which is easier to fine-tune as it would be symmetrical around the x -axis and thus more uniform. It should also required much less simulation time. If accurate results are achieved, the airfoils could be added afterwards to investigate their effect.

Another recommendation is to conduct a mesh independence study to fully secure that a mesh of high quality is used. By mesh independent study, the authors proposes that one turbulence model is chosen and the mesh is refined until it reaches the quality necessary to correctly capture the flow patterns (according to experimental data). Then, different models can be tested and a Coanda blow can be implemented. Such working procedure might be more time demanding, since the time required to acquire a good enough mesh might be very large, but on the other hand give better prerequisites to investigate the difference between the turbulence models.

Additionally, the jet was in this paper approximated with a uniform velocity boundary condition, it could be the case that this approximation is invalid and a more well-suited and realistic boundary condition would be that of a fully developed velocity profile.

Lastly, it could be that the use of wall-functions (which in this paper were used for the $k - \omega$ and $k - \omega$ -SST models) does not work well for this case. A better solution could be to skip wall-functions altogether, this would result in a much more dense mesh with a lot more cells since the boundary layer would have to be modelled and have a low y^+ near the walls. But, if this results in a more realistic result it could be worth it.

Bibliography

- [1] Energiläget 2017, Statens energimyndighet, Stockholm 2017.
- [2] Lo , K. P. Eaton, J., Flow separation control for robust conical diffuser design. Stanford University, 2012.
- [3] Versteeg, H. K. Malalasekera, W. An introduction to computational fluid dynamics, Pearson, London, 1995.
- [4] David C. Wilcox, Turbulence Modeling for CFD (Third Edition), D C W Industries, La Canada 2006.
- [5] F. Moukalled, L. Mangani, M. Darwish, The Finite Volume Method in Computational Fluid Dynamics, Springer International Publishing, Cham 2015.
- [6] L. Davidsson, Fluid mechanics, turbulent flow and turbulence modeling, Division of Fluid Dynamics Department of Applied Mechanics, Chalmers University of Technology, Göteborg 2017.
- [7] Trancossi, Michele: An Overview of Scientific and Technical Literature on Coanda Effect Applied to Nozzles, SAE Technical Papers, 2011-01-2591, 2011.
- [8] C. Lubert: On Some Recent Applications of the Coanda Effect, International Journal of Acoustics and Vibration, Vol. 16, No. 3, pp. 144-153, 2011.
- [9] J. B. Freund and M. G. Mungal: Drag and wake modification of axisymmetric bluff bodies using Coanda blowing, Journal of Aircraft, Vol. 31, No. 3, pp. 572-578, 1994.
- [10] D. S. Kumar and K. L. Kumar: Effect of Swirl on Pressure Recovery in Annular Diffusers, Journal of Mechanical Engineering Science, Vol. 22, No. 6, pp 305-313, 1980.
- [11] C.D.Argyropoulos N.C. Markatos: Recent advances on the numerical modelling of turbulent flows, Applied Mathematical Modelling, Vol 39, No. 2, pp 693-732, 2015.
- [12] Y.Zhiyin, Large eddy simulation: Past, present and the future, Chinese Journal

- of Aeronautics, Vol. 28, No. 1, pp 11-24, 2015.
- [13] M. Kuron, 3 Criteria for Assessing CFD Convergence 2015.
- [14] C. R. Maliska, E.E. Paladino, F. Saltara, B.A. Contessi, A. Ataidés, V. Girardi Silva, A comparison of turbulence models for the computation of a deattached flow around a a square cylinder CFD Lab-SINMEC, Federal University of Santa Catarina, Florianópolis, SC, Brazil, 2012.
- [15] L. Davidsson, The SAS model: A turbulence model with controlled modelled dissipation, Chalmers University of Technology, Göteborg 2007.
- [16] K. R. Sreenivas, Free shear flows, JNCASR Bangalore
- [17] A. M. Karpinska: CFD-aided modelling of activated sludge systems - A critical review, Water Research, Vol. 88, pp 861-879, 2016.
- [18] E. Saadati, Introduction to FLUENT training, Sharif University of Technology 2009.
- [19] Lo,Eaton and Elkins, Boundary layer separation control using a step-wall conical diffuser with an annular inlet, Department of Mechanical Engineering, Stanford University 2011.
- [20] F. Liu: A Thorough Description Of How Wall Functions Are Implemented In OpenFOAM, Chalmers university of techology, Göteborg 2017.
- [21] V. Dragan, Aerodynamic and Acoustic Parameters of a Coandă Flow – a Numerical Investigation, Department of Aerospace Sciences Faculty of Aerospace Engineering, Politehnica University of Bucharest, Romania 2012.
- [22] Mohs en Jahanmiri, Active Flow Control: A Review, Division of Fluid Dynamics, Department of Applied Mechanics , Chalmers University of Technology , Göteborg 2010.
- [23] J.H Ferziger ,M. Peric , Computational Methods for Fluid Dynamics, Springer, Stanford, 2002.

A

Appendix

A.1 y-Plus values

Table A.1: y^+ values

y+	Geometry 1						Geometry 2					
	<i>Medium mesh</i>		<i>Large mesh</i>		<i>Medium mesh</i>		<i>Large mesh</i>		<i>Medium mesh</i>		<i>Large mesh</i>	
	min	max	mean	min	max	mean	min	max	mean	min	max	mean
Patch												
O-RING	9.599	10.018	9.817	1.694	1.740	1.716	0.567	1.117	0.843			
MIDDLESEC	0.418	31.890	10.632	0.083	4.109	0.904	0.021	7.680	1.543			
O-RING_2	16.263	19.853	18.287	3.876	129.928	49.862	14.176	35.870	24.073			
END	0.001	1.192	0.197	0.012	1.687	0.329	0.006	7.607	0.314			
JET	0.171	1.692	0.726	0.234	29.575	9.760	1.544	12.826	6.433			
MIDDLESEC_2	0.865	5.973	2.198	17.251	85.061	41.670	5.106	12.693	9.845			
AIRFOIL3D	0.342	55.652	8.585	0.849	242.202	48.754	0.472	59.837	17.839			
OUTER_SHELL	0.082	7.748	2.020	0.049	6.382	1.040	0.005	1.648	0.065			
NEW_OUTERSHELL							0.009	3.159	0.531			

A.2 Boundary values

Table A.2: Boundary conditions used

height	Turbulent dissipation, ϵ		Turbulent kinetic energy, k		Turbulent viscosity, ν_t		Sub grid scale viscosity, ν_{sgs}	Specific dissipation rate ω		Pressure, p		Velocity, U	
Patches	k- ω /k- ω -SST	k- ω -SST-SAS	k- ω /k- ω -SST	k- ω -SST-SAS	k- ω /k- ω -SST	k- ω /k- ω -SST	k- ω /k- ω -SST	k- ω /k- ω -SST	k- ω /k- ω -SST	k- ω /k- ω -SST	k- ω /k- ω -SST	k- ω /k- ω -SST	k- ω /k- ω -SST
Inlet	fixedValue	fixedValue	fixedValue	fixedValue	calculated	calculated	calculated	fixedValue	fixedValue	zeroGradient	zeroGradient	fixedValue	fixedValue
Outlet	zeroGradient	zeroGradient	zeroGradient	zeroGradient	calculated	calculated	calculated	zeroGradient	zeroGradient	fixedValue	fixedValue	inletOutlet	inletOutlet
Other patches	epsilonWallFunction	epsilonWallFunction	kqRWallFunction	zeroGradient	nutWallFunction	nutWallFunction	zeroGradient	omegaWallFunction	zeroGradient	zeroGradient	zeroGradient	zeroGradient	fixedValue

A.3 OpenFOAM code for swirl implementation

Credits for the code goes to Ardalan Javadi.

Filename: rotatingWallTangentialVelocityFvPatchVectorField.C

```

/*-----*\
=====
\\      /  F i e l d           | foam-extend: Open Source CFD
\\      /  O p e r a t i o n   |
\\      /  A n d                | For copyright notice see file Copyright
  \\    /  M a n i p u l a t i o n |
-----*/

License
This file is part of foam-extend.

foam-extend is free software: you can redistribute it and/or modify it
under the terms of the GNU General Public License as published by the
Free Software Foundation, either version 3 of the License, or (at your
option) any later version.

foam-extend is distributed in the hope that it will be useful, but
WITHOUT ANY WARRANTY; without even the implied warranty of
MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
General Public License for more details.

You should have received a copy of the GNU General Public License
along with foam-extend. If not, see <http://www.gnu.org/licenses/>.

\*-----*/

#include "rotatingWallTangentialVelocityFvPatchVectorField.H"
#include "addToRunTimeSelectionTable.H"
#include "volFields.H"
#include "surfaceFields.H"

// * * * * *

namespace Foam
{
// * * * * * Constructors * * * * *

rotatingWallTangentialVelocityFvPatchVectorField::rotatingWallTangentialVelocityFvPatchVectorField
(
    const fvPatch& p,
    const DimensionedField<vector, volMesh>& iF
)
:
    fixedValueFvPatchField<vector>(p, iF),
    origin_(vector::zero),
    axis_(vector::zero),
    omega_(0),
    axialVelocity_(0),
    radialVelocity_(0),
    tangVelocity_(0)
{}

rotatingWallTangentialVelocityFvPatchVectorField::rotatingWallTangentialVelocityFvPatchVectorField
(
    const rotatingWallTangentialVelocityFvPatchVectorField& ptf,
    const fvPatch& p,
    const DimensionedField<vector, volMesh>& iF,
    const fvPatchFieldMapper& mapper

```

```

)
:
    fixedValueFvPatchField<vector>(ptf, p, iF, mapper),
        origin_(ptf.origin_),
        axis_(ptf.axis_),
        omega_(ptf.omega_),
//////////

        axialVelocity_(ptf.axialVelocity_),
        tangVelocity_(ptf.tangVelocity_),
        radialVelocity_(ptf.radialVelocity_)
//////////
{}

rotatingWallTangentialVelocityFvPatchVectorField::rotatingWallTangentialVelocityFvPatchVectorField
(
    const fvPatch& p,
    const DimensionedField<vector, volMesh>& iF,
    const dictionary& dict
)
:
    fixedValueFvPatchField<vector>(p, iF),
    origin_(dict.lookup("origin")),
    axis_(dict.lookup("axis")),
    omega_(readScalar(dict.lookup("omega"))),
//////////
    axialVelocity_(readScalar(dict.lookup("axialVelocity"))),
    tangVelocity_(readScalar(dict.lookup("tangVelocity"))),
    radialVelocity_(readScalar(dict.lookup("radialVelocity")))
//////////
{
    // Evaluate the wall velocity
    updateCoeffs();
}

rotatingWallTangentialVelocityFvPatchVectorField::rotatingWallTangentialVelocityFvPatchVectorField
(
    const rotatingWallTangentialVelocityFvPatchVectorField& pivpvf
)
:
    fixedValueFvPatchField<vector>(pivpvf),
    origin_(pivpvf.origin_),
    axis_(pivpvf.axis_),
    omega_(pivpvf.omega_),
//////////
    axialVelocity_(pivpvf.axialVelocity_),
    tangVelocity_(pivpvf.tangVelocity_),
    radialVelocity_(pivpvf.radialVelocity_)
//////////
{}

s
rotatingWallTangentialVelocityFvPatchVectorField::rotatingWallTangentialVelocityFvPatchVectorField
(
    const rotatingWallTangentialVelocityFvPatchVectorField& pivpvf,
    const DimensionedField<vector, volMesh>& iF
)
:
    fixedValueFvPatchField<vector>(pivpvf, iF),
    origin_(pivpvf.origin_),
    axis_(pivpvf.axis_),
    omega_(pivpvf.omega_),
//////////

    axialVelocity_(pivpvf.axialVelocity_),
    tangVelocity_(pivpvf.tangVelocity_),
    radialVelocity_(pivpvf.radialVelocity_)
//////////

```

A. Appendix

```
{

// * * * * * Member Functions * * * * * //

void rotatingWallTangentialVelocityFvPatchVectorField::updateCoeffs()
{
    if (updated())
    {
        return;
    }

    // const scalar t = this->db().time().timeOutputValue();
    // const scalar axialVelocity = axialVelocity_->value(t);
    // const scalar radialVelocity = radialVelocity_->value(t);
    // Calculate the rotating wall velocity from the specification of the motion
    // vectorField Up = (-omega_)*((patch().Cf() - origin_) ^ (axis_/mag(axis_)));

    // Remove the component of Up normal to the wall
    // just in case it is not exactly circular
    // vectorField n = patch().nf();
    // vectorField::operator=(Up - n*(n & Up));

    vector hatAxis = axis_/mag(axis_);

    const vectorField r(patch().Cf() - origin_);
    const vectorField d(r - (hatAxis & r)*hatAxis);

    //tmp<vectorField> tangDir
    vectorField tangDir
    (
        (patch().Cf() - origin_) ^ (axis_/mag(axis_))
    );
    operator==(-omega_*tangDir + -tangVelocity_*tangDir/mag(tangDir) + hatAxis*axialVelocity_ +
    radialVelocity_*d/mag(d));
    fixedValueFvPatchVectorField::updateCoeffs();
}

void rotatingWallTangentialVelocityFvPatchVectorField::write(Ostream& os) const
{
    fvPatchVectorField::write(os);
    os.writeKeyword("origin") << origin_ << token::END_STATEMENT << nl;
    os.writeKeyword("axis") << axis_ << token::END_STATEMENT << nl;
    os.writeKeyword("omega") << omega_ << token::END_STATEMENT << nl;
    os.writeKeyword("axialVelocity") << axialVelocity_ << token::END_STATEMENT << nl;
    os.writeKeyword("radialVelocity") << radialVelocity_ << token::END_STATEMENT << nl;
    os.writeKeyword("tangVelocity") << tangVelocity_ << token::END_STATEMENT << nl;
    writeEntry("value", os);
}

// * * * * * //

makePatchTypeField
(
    fvPatchVectorField,
    rotatingWallTangentialVelocityFvPatchVectorField
);

// * * * * * //

} // End namespace Foam

// ***** //

Filename: rotatingWallTangentialVelocityFvPatchVectorField.H
```

```

/*-----*\
===== |
\\      / F i e l d      | foam-extend: Open Source CFD
\\      / O p e r a t i o n |
\\      / A n d           | For copyright notice see file Copyright
\\\/    M a n i p u l a t i o n |
-----*/

License
  This file is part of foam-extend.

  foam-extend is free software: you can redistribute it and/or modify it
  under the terms of the GNU General Public License as published by the
  Free Software Foundation, either version 3 of the License, or (at your
  option) any later version.

  foam-extend is distributed in the hope that it will be useful, but
  WITHOUT ANY WARRANTY; without even the implied warranty of
  MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
  General Public License for more details.

  You should have received a copy of the GNU General Public License
  along with foam-extend. If not, see <http://www.gnu.org/licenses/>.

Class
  Foam::rotatingWallTangentialVelocityFvPatchVectorField

Description
  Foam::rotatingWallTangentialVelocityFvPatchVectorField

SourceFiles
  rotatingWallTangentialVelocityFvPatchVectorField.C

/*-----*\

#ifndef rotatingWallTangentialVelocityFvPatchVectorField_H
#define rotatingWallTangentialVelocityFvPatchVectorField_H

#include "fixedValueFvPatchFields.H"

// * * * * * //

namespace Foam
{
/*-----*\
                                   Class rotatingWallTangentialVelocityFvPatch Declaration
/*-----*\

class rotatingWallTangentialVelocityFvPatchVectorField
:
public fixedValueFvPatchVectorField
{
    // Private data

    //- Origin of the rotation
    vector origin_;

    //- Axis of the rotation
    vector axis_;

    //- Rotational speed
    scalar omega_;

    //- Axial Velocity
    scalar axialVelocity_;

    //- Radial Velocity
    scalar radialVelocity_;

```

A. Appendix

```
        //- Tangential Velocity
        scalar tangVelocity_;

public:

    //- Runtime type information
    TypeName("rotatingWallTangentialVelocity");

    // Constructors

    //- Construct from patch and internal field
    rotatingWallTangentialVelocityFvPatchVectorField
    (
        const fvPatch&,
        const DimensionedField<vector, volMesh>&
    );

    //- Construct from patch, internal field and dictionary
    rotatingWallTangentialVelocityFvPatchVectorField
    (
        const fvPatch&,
        const DimensionedField<vector, volMesh>&,
        const dictionary&
    );

    //- Construct by mapping given rotatingWallTangentialVelocityFvPatchVectorField
    // onto a new patch
    rotatingWallTangentialVelocityFvPatchVectorField
    (
        const rotatingWallTangentialVelocityFvPatchVectorField&,
        const fvPatch&,
        const DimensionedField<vector, volMesh>&,
        const fvPatchFieldMapper&
    );

    //- Construct as copy
    rotatingWallTangentialVelocityFvPatchVectorField
    (
        const rotatingWallTangentialVelocityFvPatchVectorField&
    );

    //- Construct and return a clone
    virtual tmp<fvPatchVectorField> clone() const
    {
        return tmp<fvPatchVectorField>
        (
            new rotatingWallTangentialVelocityFvPatchVectorField(*this)
        );
    }

    //- Construct as copy setting internal field reference
    rotatingWallTangentialVelocityFvPatchVectorField
    (
        const rotatingWallTangentialVelocityFvPatchVectorField&,
        const DimensionedField<vector, volMesh>&
    );

    //- Construct and return a clone setting internal field reference
    virtual tmp<fvPatchVectorField> clone
    (
        const DimensionedField<vector, volMesh>& iF
    ) const
    {
        return tmp<fvPatchVectorField>
        (
            new rotatingWallTangentialVelocityFvPatchVectorField(*this, iF)
        );
    }
}
```

```
// Member functions

// Access functions

//- Return the origin of the rotation
const vector& origin() const
{
    return origin_;
}

//- Return the axis of the rotation
const vector& axis() const
{
    return axis_;
}

//- Return the rotational speed
const scalar& omega() const
{
    return omega_;
}

//- Return non-const access to the origin of the rotation
vector& origin()
{
    return origin_;
}

//- Return non-const access to the axis of the rotation
vector& axis()
{
    return axis_;
}

//- Return non-const access to the rotational speed
scalar& omega()
{
    return omega_;
}

//- Update the coefficients associated with the patch field
virtual void updateCoeffs();

//- Write
virtual void write(Ostream&) const;
};

// * * * * *
} // End namespace Foam
// * * * * *

#endif

// ***** //
```

A.4 Implementation in OpenFOAM

A.4.1 Implementation of turbulence models

Three turbulence ($k - \omega$, $k - \omega$ -SST and $k - \omega$ -SST SAS) models were used during the simulations; each for the two geometrical setups in addition to with and without Coanda blowing. The implementation of those models is described below.

A.4.1.1 $k - \omega$ and $k - \omega$ -SST

The usage and implementation of those two turbulence models was very similar, as they are both based on the addition of a "RASProperties" file in the */constant* dictionary. The chosen *RASProperties* file contained all the, for this project, relevant coefficients and constants used in $k - \omega$ and $k - \omega$ -SST turbulence modelling. To change between the two is then the trivial matter of changing name in the *RASProperties* file between *kOmega* and *kOmegaSST*.

A.4.1.2 $k - \omega$ -SST SAS

The implementation of the SAS model is slightly more complex than the previous two, mainly due to the fact that it requires some improvements of the mesh. The mesh used in the simulation with $k - \omega$ and $k - \omega$ -SST as the chosen turbulence models contained roughly 300 000 cells. SAS, on the other hand, requires a completely different y^+ (see the section 3.1.2 for a description of why). Therefore, the SAS turbulence model was done after the two previous models were simulated, as the regions with an insufficiently small y^+ was found using the command "yPlus-latestTime" while those simulations were running. By doing so, the mesh could be refined in the relevant areas to eventually fulfil the aim to have

- Between 1-2 million cells
- $y^+ \approx 1$ and $y^+ \not\approx 4$.

Once those criteria were met for the mesh, the actual SAS model could be implemented with the changes to the variables in the "0" directory as stated in the previous section and the addition of *LESProperties* to the */constant* folder.

A.4.2 Other noteworthy changes to the OpenFOAM settings for the case

An important change that was made to make the solution converge was done in the *fvSchemes* folder. Due to the complexity of the mesh, and consequently difficulties in achieving a good aspect ratio between cells, a change in the number of correctors proved to be a successful way to achieve simulation stability. More precisely, the correctors used were *nonOrthogonalCorrectors*, *nCorrectors* and *nOuterCorrectors*. Additionally, adding relaxation factors also helped to further stabilise the simulation.

The number of correctors (of all three sorts) proved to be an effective way to speed up the simulation, and simultaneously have control over the Courant number and residuals. Additionally, the time step was slowly increased throughout the simulation, starting off at e-06 and ending at either e-02 or e-03 depending on the mesh and turbulence model. As a too rapid of a change in time step, as well as the numbers of correctors, caused the Courant number to increase rapidly, a slow step by step protocol was used to eventually reach the settings the majority of the simulation was run at.

For the simulations where Coanda blowing was used (its purpose and effect is described in section 3.1.3.1 and 2.5.2 respectively), implementation in OpenFOAM was done by simply setting a boundary condition at the jet.